;login:

The USENIX Association Newsletter

Volume 15, Number 1 **January/February 1990**

CONTENTS

The closing date for submissions for the next issue of *;login:* is February 23, 1990.

USENIX THE PROFESSIONAL AND TECHNICAL
UNIX® ASSOCIATION

NOTICE

;login: (ISSN 1044-6397) is published bi-monthly by the USENIX Association,

The USENIX Association is a not-for-profit organization of those interested in UNIX[†] and UNIX-like systems. It is dedicated to fostering and communicating the development of research and technological information and ideas pertaining to advanced computing systems, to the monitoring and encouragement of continuing innovation in advanced computing environments, and to the provision of a forum where technical issues are aired and critical thought exercised so that its members can remain current and vital.

The officers of the Association are:

President	Alan G. Nemeth *agn@usenix.org*
Vice-President	Deborah K. Scherrer *scherrer@usenix.org*
Secretary	Rob Kolstad *kolstad@usenix.org*
Treasurer	Stephen C. Johnson *scj@usenix.org*
Directors	M. Kirk McKusick *mckusick@usenix.org* Sharon Murrel *eowyn@usenix.org* Michael D. O'Dell *mo@usenix.org* John S. Quarterman *jsq@usenix.org*
Executive Director	Ellie Young *ellie@usenix.org*

USENIX Association Office
2560 Ninth Street, Suite 215
Berkeley, CA 94710
(415) 528-8649
office@usenix.org

The editorial staff of *;login:* is:

Editor	Ellie Young
Managing Editor	Carolyn Carr
Copy Editor	Michelle Dominijanni
Production Editor	Tom Strong

Judith F. DesHarnais, Conference Coordinator
USENIX Conference Office
22672 Lambert Street, Suite 613
El Toro, CA 92630
(714) 588-8649
judy@usenix.org

John L. Donnelly, Tutorial & Exhibit Manager
USENIX Exhibit Office
5398 Manhattan Circle
Boulder, CO 80303
(303) 499-2600
johnd@usenix.org

Contributions Solicited

Members of the UNIX community are encouraged to contribute articles to *;login:*. Contributions may be sent electronically to *login@usenix.org* or through the U.S. mail to the Association office. The USENIX Association reserves the right to edit submitted material.

;login: is produced on UNIX systems using *troff* and a variation of the –me macros. Contributions should be in *n/troff* input format, using any macro package.

UUNET Subscriptions

UUNET Communications
3110 Fairview Park Drive, Suite 570
Falls Church, VA 22042
(703) 876-5050
uunet-request@uunet.uu.net

Acknowledgments

The Association uses a Sun[‡] 3/180S running SunOS for support of office and membership functions, preparation of *;login:*, and other Association activities. Connected to the Sun is a QMS Lasergrafix[*] 800 Printer System donated by Quality Micro Systems of Mobile, Alabama. It is used for general printing and draft production of *;login:* and *Computing Systems* with *ditroff* software provided by mt Xinu. The membership and mailing lists are maintained using Sybase.[#]

[†]UNIX is a registered trademark of AT&T.
[‡]SUN is a trademark of Sun Microsystems, Inc.
[*]Lasergrafix is a trademark of Quality Micro Systems.
[#]Sybase is a registered trademark of Sybase, Inc.

Call for Papers: Summer 1990 USENIX Conference

Anaheim, CA, June 11-15, 1990

USENIX continues to seek papers describing new and interesting work. However, the Summer 1990 Technical Conference also seeks to include papers that emphasize retrospectives, analyses of tradeoffs, and critical thinking about where we are, how we got here, and why we're here. Thus, the theme is:

Beyond Mere Data:
Perspective, Insight, Understanding.

Some sessions will follow the normal 3-paper format, with questions following each talk. In other sessions, the speakers will form a panel, following the talks, first to compare approaches, and then to take questions from the audience. In some cases, other experts may be added to the panel to broaden the discussion. Especially desirable are sessions where several important different viewpoints are represented, and proposals for such sessions are welcome.

Appropriate topics include, but are not limited to:

Software release systems
User interfaces, windowing, graphics
Compilers, debuggers, tools, run-time issues
File systems
Distributed systems
UNIX kernel approaches
Fault-tolerancy, reliability, or security
Computer architectures that stretch UNIX

We will accept full papers, but require at least an abstract and outline, in a form that gives the committee confidence in the final paper. A submission should be 2-3 typewritten pages and include the following:

1. Author names, addresses, telephone numbers, and E-mail addresses.

2. Abstract: 100-300 words (half a page) to be included in the final paper.

3. Outline: 1.5-2.5 pages, giving the major headings of the paper, plus a few sentences per section that give the major points that will be covered in that section in the final paper.

The following is a sample outline, which is not necessarily appropriate for all papers, but which illustrates the important topics. The purpose of an outline should be to convince the committee that something interesting and important will be said in the final paper.

1. Introduction
   - Background.
     Introduce the problem to be solved; why is it important?
     Reference previous work; make sure the committee knows the wheel is not being reinvented.
2. How We Solved the Problem
   - More details on the problem and its issues.
   - Design decisions and tradeoffs, and why they were made.
   - Implementation issues.
3. Evaluation
   - Data, on performance, effort required.
   - How well does it work?
   - What would we do differently?
   - If it failed, why? and what can we learn from it?
4. Conclusion
   - Summarize the paper, emphasizing why it is important, and what was learned.
5. References
   - List at least a few key references, preferably to other people's work.

The final paper should retain the 100-300 word abstract, include illustrations (where needed), and citations to relevant literature. Only previously unpublished submissions will be considered, although "retrospective" papers may describe work done years ago. Thinly-disguised product announcements are rarely accepted. Final papers should contain 8-12 pages of single spaced typeset materials. All final papers must be submitted in a camera-ready format or electronic format (*troff*-ms if possible). Typewritten or dot-matrix output is not acceptable. For authors without access to a laser printer or typesetter, appropriate facilities will be provided by the program chair.

Please submit abstracts with outline and proposals for sessions as soon as possible, and mail one hard copy and one electronic copy to the addresses below. The final deadline for receipt of submissions is **February 7, 1990.** Abstracts received after this deadline will not be considered. Notification of acceptance or rejection will be made by March 9, 1990. Final camera-ready papers are due by April 17, 1990.

John R. Mashey
Anaheim USENIX Technical Program
MIPS Computer Systems
930 Arques Ave
Sunnyvale, CA 94086

Internet: anaheim@mips.com
UUCP: uunet!mips.com!anaheim

Phone: (408) 991-0253
FAX: (408) 720-9809

Please include your physical and electronic mail address in all correspondence.

Program Committee:

John R. Mashey, (Chair)
MIPS Computer Systems
Clem Cole
Cole Computer Consulting
Doug Comer
Purdue University
Tom Ferrin
Univ. of CA - San Francisco
James Gettys
Digital Equipment Corp.
Lori Grob
Chorus Systems
Douglas P. Kingston III
Morgan Stanley & Co., Inc.
Heinz Lycklama
Interactive Systems Corp.
M. Douglas McIlroy
AT&T Bell Laboratories
Joe Moran
Legato Systems, Inc.
Pat Parseghian
Princeton University
Lawrence Rosler
Hewlett Packard
Bill Shannon
Sun Microsystems, Inc.

C++ Conference

Marriott Hotel, San Francisco, CA, April 9-11, 1990

USENIX is pleased to announce its 1990 C++ Conference. It will be devoted exclusively to C++ and will offer an intensive three day program bringing together in-depth tutorials along with Technical Sessions covering a broad spectrum of work.

April 9 – Full Day Tutorials

Tutorial Program

An Introduction To C++

Robert Murray, AT&T Bell Laboratories

A survey of the main features of C++ (including features added in Version 2.0) will be presented, with some short examples on how to use the features effectively. Most use of C++ falls into one of three flavors: a better C, data abstraction, and object-oriented programming. We will examine these flavors, starting with the features and paradigms that are closest to C, and progressing to the more ambitious (and potentially more powerful) features. We'll also discuss the relationship between ANSI C, C++ Version 1.2, and C++ Version 2.0.

Effective Use of C++

Andrew Koenig, AT&T Bell Laboratories

A review of the central concepts of C++, the ways in which the language supports those concepts, and a detailed tour through several complete programming examples. This tutorial will emphasize "how to use it well" rather than "what the features are." Attendees are presumed to be capable of looking up details of syntax and semantics themselves.

A Tour of Cfront: Cfront 2.0 Internals

Stanley Lippman, AT&T Bell Laboratories

This tutorial surveys selected internal data structures and algorithms used by Cfront for the implementation of such C++ language features as multiple inheritance, virtual base classes, virtual functions, and the static initialization and deallocation of objects. We will attempt to make sense of the generated intermediate C code in light of these structures. Examples of both effective and ineffective coding styles will be discussed.

Half Day Tutorials

Using C++ on the Macintosh

Bill Gibbons, Consultant, & Ken Friedenbach, Cadence Design Systems

Macintosh Programmer's Workshop (MPW) C++ is an adaptation of the AT&T C++ Language Translation System version 2.0. This tutorial will provide information regarding the MPW C++ language, as well as information about libraries, debuggers, browsers, and other software development support tools on the Macintosh. Topics include: overview of MPW C++ language features, support for the Macintosh toolbox and operating system, support for the Macintosh memory model, and language support for MacApp. The tutorial will explain the development of sample Macintosh applications and MPW tools using C++, as well as covering useful programming techniques and common errors to avoid.

Using C++ with MacApp

Ken Friedenbach, Cadence Design Systems

MacApp is an extensible Macintosh Application which simplifies the task of writing a fully functional Macintosh application. Macintosh Programmer's Workshop (MPW) C++ includes features to support using C++ to develop MacApp applications. This tutorial will provide information about using MPW C++ with MacApp to implement fully functional Macintosh applications. Topics include: C++ language support for MacApp and Object Pascal, overview of the MacApp libraries, supporting multiple documents and windows, using the clipboard to support cut and paste, printing, and reading and writing document data. The tutorial will include sample application code, guidelines for creating building blocks to be shared between applications, and

advice for mixing MacApp classes with multiple inheritance classes.

April 10 & 11

Technical Sessions

Full details regarding the Technical program will be available by mid-February. Please contact the USENIX Conference office.

Submissions for papers closed Jan. 12, 1990.

Professional Development Seminars

The USENIX Association has initiated a series of Professional Development Seminars in cities throughout the United States. The seminar program is a subset of the highly acclaimed tutorial offerings presented at the Association's semi-annual Conferences. It will be presented in major metropolitan areas that are not currently scheduled for one of the Conferences.

The Seminar Program

After a successful debut in Chicago, October 30, 1989, the next program is scheduled for the Houston Marriott at the Astrodome, March 19, 1990.

The tutorials being offered provide an in-depth look at three different areas: internals; networking; and software development. Attendees needing to expand their knowledge in any of these areas will benefit from the material presented by the instructors – all are experts in their respective fields. Enrollment is limited to 150 people per tutorial to ensure a consistent level of interaction between the instructors and attendees.

The tutorials being offered in Houston are:

UNIX System V Release 4.0 Internals
– An Introduction
Steve Buroff & Mike Scheer, AT&T

UNIX Network Programming
Richard Stevens, Health Systems International

Software Development Using UNIX & C
Rob Kolstad, Sun Microsystems

For further information on the Houston Professional Development Seminar, please contact:

USENIX Tutorial Office
5398 Manhattan Circle
Boulder, CO 80303

(303) 499-2600
(303) 499-2608 (FAX)
johnd@usenix.org

Call for Papers: UNIX Security Workshop

Marriott Hotel, Portland, OR, August 27-28, 1990

The Second UNIX Security Workshop is to be held in Portland, Oregon, on Monday and Tuesday, August 27 and 28, 1990. Matt Bishop will again be chairing this workshop. It will bring together researchers in computer security dealing with UNIX and system administrators trying to use UNIX in environments where protection and security are of vital importance. It is intended to provide an environment where researchers can discuss their latest results, where researchers and practitioners can discuss the applicability of those results to practical problems, and where system administrators can share their unique solutions and techniques for dealing with problems. The topics covered by this workshop include both theoretical topics and everyday problems. We expect each participant to present unique attributes of his/her environment and/or research and contribute a short (five minute) discussion (and paper) detailing some result or solution from their environment or work.

Some topics to be considered include:

modeling the UNIX operating system theoretically
password security (password file integrity, enforcing choice of a safe password, spotting and handling crackers)
network security (problems arising from logins over an unprotected Ethernet, containing a break-in to one machine in a networked environment)
security in a distributed system or environment
file system security (auditing packages, security in an NFS environment)
computer worms, viruses, and other phenomena
new designs to obtain C-level (or better) certification
making existing UNIX systems more secure, and locating and fixing UNIX security problems
any other problem or contribution that participants make.

Workshop Format

This gathering will follow a "workshop" format rather than a "paper presentation" format. Please submit a one or two page summary describing a problem and, if you have one, a solution or if not, a possible approach or approaches which looked promising but failed (or which you have not yet tried). Also, be sure to include with your submission a set of five (or so) topics that you'd like to hear about. It is possible that some participants will not present their papers at this workshop.

The workshop chairman will collate the papers to schedule sessions which have appropriate audiences. It is anticipated that some sessions will include all participants though others may require breaking into smaller groups. Send your submissions to the address below by **May 22, 1990**.

For further information, contact:

Matt Bishop
Dept. of Mathematics and Computer Science
Bradley Hall
Dartmouth College
Hanover, NH 03755

(603) 646-3267
decvax!dartvax!Matt.Bishop
Matt.Bishop@dartmouth.edu

Call for Papers: Mach Workshop

Radisson Hotel, Burlington, VT, October 4-5, 1990

The use of Mach in what has traditionally been the UNIX community is growing as DARPA and OSF increase their Mach-related activities and more vendors are supporting Mach on a variety of platforms. Because Mach itself is changing rapidly and there hasn't been any convenient mechanism for communication among developers, the USENIX Association is pleased to sponsor its first Mach workshop, in which researchers, vendors, and users can share results of Mach-related development work and status reports on work-in-progress.

The workshop will be oriented towards those who have actually worked with Mach or have done Mach-based applications development, and will not be tutorial in nature. The program will consist largely of refereed papers and panels. Abstracts of 350-700 words should be sent to the program chair at the address below (those submitting hardcopy abstracts should send five copies). The deadline for submissions is June 22, 1990. All submissions will be acknowledged. Authors will be notified by July 20, 1990, and full papers will be required by August 27, 1990.

For further information about the workshop, contact the program chair:

Melinda Shore
mt Xinu
2560 Ninth St., Suite 312
Berkeley, CA 94710

(415) 644-0146
shore@mtxinu.com

Program Committee:

Alan Langerman, Encore Computer Corporation
Douglas Orr, Carnegie-Mellon University
Homayoon Tajalli, Trusted Information Sys.
Avadis Tevanian, NeXT, Inc.

Request for Proposals to Chair the Large Installation Systems Administration Workshop (LISA)

The USENIX Association is seeking proposals from people interested in chairing its fourth LISA workshop, to be held sometime in the Fall of 1990.

We are seeking an energetic person with the following qualifications:

- Good administrative skills
- Extensive experience in the administration of large installation systems
- Good public speaking skills
- Knowledge of what are the timely and appropriate topics in the field
- Ability to solicit good panel members and appropriate speakers
- Attendance at one previous LISA workshop

Proposals should be brief (one page) and might include the following:

- Statement of Purpose (e.g., why should we have another one?)
- Form of submissions: abstracts, extended abstracts or full papers?
- Format (e.g., two days of technical sessions, panel sessions, etc.)
- List of topics to be addressed
- Special features: such as having tutorials, BOFs
- List of potential program committee members and/or a co-chair*
- Brief biography

Proposals are subject to approval by the board of directors. Details concerning the schedule, site and call for papers will be worked out with the Association after the appointment of the chair has been made.

Proposals are due: **February 15, 1990.**

Please address all inquiries and proposals to the Association's Executive Director, Ellie Young (ellie@usenix.org).

* While most USENIX workshops have an individual chair, proposals requesting a co-chair and/or small program committee are welcome.

Call for Papers: EUUG Conference

October 22-26, 1990, Nice, France

The Autumn 1990 European UNIX systems User Group Technical Conference will be held October 22-26, 1990, at the Nice Acropolis, Nice, France. Technical tutorials on UNIX and closely related subjects will be held on October 22-23, followed by the three day conference with a commercial exhibition.

Call for Papers

The EUUG invites papers from those wishing to present their work. Full papers or extended abstracts must be submitted. All submitted papers will be refereed and judged with respect to their quality, originality, and relevance.

Suggested subject areas include, but are not limited to:

- Software Management for large projects
  - Configuration Management
  - Maintenance Management
  - Update and Release control
- OSI and OSI application on a UNIX platform
- System Administration in a Hetrogeneous environment
- Security and Audit
  - Secure UNIX
  - Securing existing systems
- UNIX in non-English speaking environments
- User Interface Management Systems

Submissions from students are particularly encouraged under the EUUG Student Encouragement Scheme, details of which are available from the EUUG Secretariat.

Important Dates

Abstract deadline	April 30, 1990
Acceptance notification	May 10, 1990
Final paper received	June 30, 1990
Student Grant Applications deadline	Sept. 1, 1990

Method of Submission

Full papers or extended abstracts **must** be submitted by post to the EUUG Secretariat and, if possible, in electronic form to *euug-nice@eu.net*. All submissions will be acknowledged by return of post.

Guidance to Authors

A copy of the EUUG guidance to Authors will be sent automatically to everybody that submits a paper. It will also be printed in the Spring edition of the EUUG newsletter.

Tutorial Solicitation

Tutorials are an important part of the EUUG's biannual events providing detailed coverage of a number of topics. Past tutorials have been taught by leading experts.

We are keen to provide classes to all levels. Those interested in offering a tutorial should contact the EUUG Tutorial Executive as soon as possible.

Additional Information

We will be pleased to provide advice to potential speakers, and can be contacted at the addresses below.

If you wish receive further information about this and future EUUG events, please write, or send electronic mail, to the Secretariat.

EUUG Secretariat
Owles Hall
Owles Lane
Buntingford, Herts SG9 9PL UK

Phone: (+44) 763 73039
Fax: (+44) 763 73255
Email: euug@eu.net

Conference & Tutorial Executive
Neil Todd
GiD Ltd
Email: neil@gid.co.uk

Program Chair
Johan Helsingius
OY Penetron Ab
Email: julf@fuug.fi

Program Committee

Pierre Louise Neumann	INRIA France
Dr. Elod Knuth	MTA SZTAKI, Hungary
Daniel Klein	CMU, USA

Call for Papers: UKUUG Conference

"UNIX – The Legend Evolves" London, England, July 9-13, 1990

The UKUUG Summer 1990 Technical Conference will take place at the Royal Lancaster Hotel in London on July 11-13, preceded by two days of advanced tutorials beginning on Monday, July 9. The three day conference offers presentations by world renowned exponents of UNIX related systems supported by "work in progress" and "poster" sessions.

Program and Speakers

Brian Kernighan will speak about the progress which has been made in the development of programming environments, languages, tools, and even "methodologies," discussing how past experiences will influence their future evolution. Ken Thompson, Rob Pike, and Dave Presotto have been conducting research into a successor to UNIX entitled "Plan 9 from Bell Labs." The result is a novel and innovative distributed system which runs counter to the popular trend in computing environments, namely workstations connected by local area networks. Each will speak on his contribution to the proposal of a new system based on clusters of file servers and execute servers connected by high speed networks.

Other noteworthy presenters include Jon Bentley, Doug Comer, Piers Dick-Lauder, Dennis Ritchie, Stu Feldman, James Gosling, Mike Karels, Kirk McKusick, and Andy Tanenbaum.

Call for Papers

Technical papers are sought in **all** areas of UNIX-related research and development, which includes work associated with programming languages like C and C++. Accepted papers will be published in the conference proceedings and presented during the three days of technical sessions.

Of particular interest, but not confined to this area alone, are new and interesting submissions about distributed environments: file systems, data bases, execution and load balancing, kernels, windowing systems, etc. Also welcome are papers concerning tools, "little languages," networks, security, system administration, and standards.

Method of Submission and Important Dates

Extended abstracts of between two and four pages in length are invited describing the nature of the work along with a summary of results and conclusions. All abstracts must be submitted to the Program Director who is also willing to provide advice to potential speakers. High quality original work will be regarded much more favorably than re-runs of previous papers. **Submissions from students are encouraged and will be particularly favored.**

Receipt of Extended Abstracts:	**Jan. 26, 1990**
Acceptance Notifications:	**Jan. 31, 1990**
Final Paper Due:	**April 6, 1990**

Solicitation for Other Contributions

The one day advanced tutorials play an important role in UKUUG's activities and so experts are encouraged to discuss topics and subject matter with the Program Director. For the first time at a UKUUG conference, a "work in progress" session will be held where 10-15 minute slots are available, with priority being given to reports of student projects and research activity. For those preferring one-to-one contact, "poster" sessions are available in the terminal room next to the auditorium for technical sessions.

Contact:

Program Director
Sunil K. Das
City University London
Computer Science Department
Northampton Square EC1V 0HB UK
(01) 253-4399 x3725
sunil@cs.city.ac.uk

Long-Term Calendar of UNIX Events†

1990 Feb 14	UKUUG Sys. Admin. Workshop	Inst. of Ed, London, UK
1990 Mar 5-6	X3J11	New York, NY
1990 Mar 26-29	DECUS	Vasteras, Sweden
1990 Mar 26-30	AFUU	Paris, France
1990 Apr 9	POSIX APP Workshop	NIST; Gaithersburg, MD
1990 Apr 9-11	USENIX C++ Conference	San Francisco, CA
1990 Apr 23-27	EUUG	Munich, Germany
1990 Apr 23-27	IEEE 1003	Salt Lake City, UT
1990 May 7-11	DECUS	New Orleans, LA
1990 May 17	U & Parallel Systems, NLUUG	Ede, Netherlands
1990 May 30-Jun 1	UNIX/90	/usr/group/cdn; Toronto, Ont.
1990 Jun 11-15	USENIX	Marriott Hotel, Anaheim, CA
1990 Jun 11-15	ISO WG15 (POSIX)	Paris, France
1990 Jul 9-11	15th JUS Symposium	Tokyo, Japan
1990 Jul 11-13	UKUUG	London, UK
1990 Jul 16-20	IEEE 1003	Danvers, MA
1990 Aug 27-28	*Security	Portland, OR
1990 Sep 25-28	AUUG	World Congress Centre, Melbourne, Aust.
1990 Oct 3-5	Internat'l S of MHS, IFIP WG 6.5	Zurich, Switzerland
1990 Oct 4-5	*Mach	Burlington, VT
1990 Oct 8-12	InterOp 90 ACE	San Jose, CA
1990 Oct 22-26	EUUG	Nice, France
1990 Oct 22-26	IEEE 1003	Seattle, WA
1990 Oct 31-Nov 1	UNIX Expo	New York, NY
1990 Nov ?	*Software Devel. Environments	West Coast, USA
1990 Nov 5-9	Computer Communication Conf.	ICCC; New Delhi, India
1990 Nov 8	Open Systems, NLUUG	Ede, Netherlands
1990 Nov 14-16	UNIX EXPO '90 UniForum	Stockholm, Sweden
1990 Nov 15	POSIX APP Workshop	NIST; Gaithersburg, MD
1990 Nov 15-16	16th JUS Symposium	Osaka, Japan
1990 Dec 4-5	JUS UNIX Fair '90	Tokyo, Japan
1990 Dec 10-14	DECUS	Las Vegas, NV
1991 Jan 7-11	IEEE 1003	Florida
1991 Jan 21-25	USENIX	Grand Kempinski, Dallas, TX
1991 Jan 22-25	UniForum	Infomart, Dallas, TX
1991 Feb	UNIX in Government	Ottawa, Ont.
1991 Feb 18-22	DECUS	Ottawa, Ont.
1991 May	UNIX 8x/etc	/usr/group/cdn; Toronto, Ont.
1991 May 6-10	DECUS	Atlanta, GA
1991 May 20-24	EUUG	Tromso, Norway
1991 Jun 10-14	USENIX	Opryland, Nashville, TN
1991 Sep 16-20	EUUG	Budapest, Hungary
1991 Dec 9-13	DECUS	Anaheim, CA
1992 Jan 20-24	USENIX	Hilton Square, San Francisco, CA
1992 Jan 21-24	UniForum	Moscone Center, San Francisco, CA
1992 Spring	EUUG	Jersey, UK
1992 May 4-8	DECUS	Atlanta, GA
1992 Jun 8-12	USENIX	Marriott, San Antonio, TX
1992 Autumn	EUUG	Amsterdam, Netherlands

† Compiled with the assistance of Alain Williams of the EUUG, John Quarterman of Texas Internet Consulting and Susanne Smith of Windsound Consulting.

* USENIX Workshops

Speakers Bureau

The USENIX Association is seeking volunteers to participate in a Speakers Bureau. The purpose of the Speakers Bureau is twofold: to provide a forum for people with expertise in various areas of UNIX to share that knowledge with others; and, to provide a source of speakers for educational groups (high schools, colleges, universities, local user groups) who could discuss a variety of UNIX-related topics in a colloquium-type setting.

We would like the Speakers Bureau to have a local flavor where the volunteers and interested groups draw from within the local/regional community. In cases where round trip travel of greater than 50 miles is involved, USENIX would provide a mileage allowance.

Before you discard this idea, thinking you are too busy, please consider volunteering to speak just once or twice during the year. We are sure you will find this a rewarding experience and a great service to the audience.

When a request for a speaker is received, the Speakers Bureau will try to match the requirements with a particular volunteer's expertise. If you are contacted, and cannot speak, please feel free to decline. We want you to accept only those invitations which you truly want to do.

Should you decide to lend your expertise to this project, please fill out the Information Sheet and return it as soon as possible. We would like to promote this concept, but need a collection of speakers before we can announce its existence to potentially interested groups.

If you do not wish to volunteer, but know of someone within the UNIX community who might be interested in participating, please fill out the relevant part of the Information Sheet, and we shall contact that person directly. If you know of a group that would benefit from some of the resources that may become available, please let us know.

John L. Donnelly
Coordinator
USENIX Speakers Bureau

Speakers Bureau Information Sheet

Name: _____ Title: _____

Organization: _____

Address: _____

_____

Phone (work): _____ (home): _____

email address: _____

Do you have any logistical limitations where/when you can speak?

Geographical area: _____ Day of week: _____

Length of time: _____ Time of day: _____

Do you have any group preferences (high schools, colleges, user groups)?

_____

_____

How often would you make yourself available to speak?

_____

What are your particular areas of expertise?

_____

_____

Give a brief description of yourself (educational background, current position, previous speaking experience, etc.)

_____

_____

Do you know of any other people who might be interested in being involved in the Speakers Bureau (in any capacity – give name[s], phone number[s])?

_____

_____

Are you interested in using the resources of the Speakers Bureau for your group or organization? If so, what topics would you be interested in?

_____

_____

Please return this form to:

Speakers Bureau
USENIX Association
5398 Manhattan Circle, Suite 201
Boulder, CO 80303

303-499-2600
303-499-2608 (FAX)
johnd@usenix.org

UUCP Project Draws Strong Response

As discussed in the last issue of *;login:*, USENIX is studying *uucp* to see whether we can help promote better communication, in a literal sense, by activities which might range from standardization up to a possible sponsored implementation.

We have received more than 30 mail messages on this topic from Europe, Australia, and the U.S. Many of these were cheers and bravos, and most requested further information. A number were from people who were actively working on *uucp* itself, or on programs that were similar to, or "better" than *uucp*. In particular, there was relevant work going on at AT&T, GNU, in Australia with ACSnet, in Great Britain with UKUUCP, and at Prime Computer in Australia, with MYL. A program called PP, which supports numerous protocols, is in the beta stage in Great Britain, and will be "openly available." Finally, Rick Adams, has been doing advanced CPR on *uucp* for years to keep *uunet* running smoothly, and has suggested that he might make this available.

There appear to be three major decision areas (battlegrounds?). One is technical – what do we want, given that we can't have everything. Some people wrote and suggested that using anything other than streams and TLI was senseless and short-sighted; others wrote that the use of streams and TLI would lock us out of a large number of smaller and older machines, and should be avoided at all cost. I personally would like to have some graphic (X-ish) administration interface so I can add a phone number without screwing up the company-wide system for days; but this limits our communication scope.

The next set of decisions has to do with the distribution of the system; will it be free, or merely cheap. The majority think it should be distributed in source code; a vocal minority paint a picture of a totally snarled net created by enthusiastic hackery by hundreds of monkeys at their terminals. Some think it should be licensed, others totally free.

The related problem is how to get it done; should USENIX endorse, support, initiate, or purchase the work? There are a lot of touchy issues here, including what the cost would be, how it would be recovered (hold on to your wallets, members!), and how USENIX would ensure that it got its money's worth (ever try to manage a software project with volunteers?).

The USENIX board of directors will be taking this topic up again at its meeting in Washington, D.C. We shall report on the outcome from that meeting in the next issue of ;login:. I haven't really done justice in this summary to some of the lengthy, thoughtful comments that we have received; thank you for the encouragement and the information. Further comments and suggestions can be sent to *scj@usenix.org* or discussed with other USENIX board members.

Steve Johnson

Book Reviews

UNIX System Software Readings
AT&T UNIX Pacific Co., Ltd.
(Englewood Cliffs, NJ: Prentice-Hall, 1988, ISBN 0-13-938358-1)

Reviewed by George W. Leach
AT&T Paradyne

This book records the presentations made by six invited speakers from AT&T at a seminar held in Japan in the summer of 1986. The seminar was sponsored by AT&T UNIX Pacific (ATTUP) in order to disseminate information concerning new technologies for System V Release 3.0 to the Asian/Pacific UNIX community. The contents of the book are as follows:

1. Larry L. Crume, *Introduction*
2. Brian W. Kernighan, *Beyond UNIX* (Keynote Speech)
3. Gilbert J. McGrath, *Streams Technology*
4. Laurence M. Brown, *Networking Architecture & Protocol*
5. Arthur L. Sabsevitz, *Distributed UNIX System – Remote File Sharing*
6. Gary L. Lindgren, *Directions in Internationalization*
7. David G. Korn, *The Shell – Past, Present, and Future*

A great deal of the material presented here has not been readily available to the general public. For example, Kernighan's presentation covers the activities of the Computer Science Research Center at Bell Labs in Murray Hill. It has been several years since information has become available on much of the UNIX-related work going on there.[1,2] However, some of the current focus is shifting away from Research UNIX and the Blit towards Plan 9 and the Gnot.[3,4,5]

The papers by McGrath, Brown, and Sabsevitz describe the System V approaches to networking and distribution. Once again, very little has been written about Streams and the Transport Level Interface (TLI) for networking with System V.[6,7] Descriptions of RFS are hard to come by as well.[8] For someone new to System V these papers make an excellent, albeit brief, introduction to these facilities.

Some of the internationalization support that is described in Lindgren's paper will be incorporated into SVR4. At the time of the seminar, the Japanese Application Environment (JAE) 1.0 was available from ATTUP. However, today it is not even the current product offering, which will be superceded by SVR4. Still, the background material on internationalization is quite relevant and makes the paper worth reading. In fact, the existence of this paper in this collection is what attracted my attention to the book.

David Korn's paper on the UNIX Shell provides a nice history of the development of everyone's favorite command interpreter as well as an overview of its functionality. And, of course, the Korn Shell is featured in the paper. There are some comparisons of the capabilities of the Bourne, C, and Korn Shells as well, but they are minimal.

The viewgraphs from the seminar presentations are included as figures for each paper. This is a nice touch. Often folks will request copies of viewgraphs from a presenter despite the existence of a paper in the proceedings.

Overall, my impression of this collection is that for the experienced UNIX programmer the material is too elementary. The topic coverage is mostly at a conceptual level and of a tutorial nature. Furthermore, the bulk of the material is specific to System V. BSD fans will not be interested. And finally, the presentations are based upon a release of System V that is about to be eclipsed. Some of the areas covered in the book are due for some new features, for example an out-of-band communications capability will be added to Streams.[9]

There is, however, some worthwhile material in this collection for UNIX folks of all persuasions. The paper by Kernighan is probably the most concise description of the work

that the folks in Murray Hill have been involved in over the past several years. The paper on internationalization does present a nice overview of the problems involved with working on software with an eye towards the world marketplace. And David Korn's paper presents a unique perspective in the command interpreter arena. The papers are compact and to the point, much like the book (182 pages). And the list price of $21.95 won't break your budget.

References

1. AT&T, "The UNIX System," AT&T Bell Laboratories Technical Journal, Vol. 63, No. 8, Part 2, October 1984.

2. Members of CSRC, Various papers describing 8th Edition UNIX, Summer 1985 USENIX Conference Proceedings, Portland OR.

3. Presotto, Dave, "Plan 9 from Bell Labs – The Network," EUUG Spring 1988 Conference Proceedings, London UK.

4. Pike, Rob, and Thompson, Ken, "Position Paper for IEEE Workshop on Operating Systems," Workshop on Workstation Operating Systems, Cambridge MA, November 5-6, 1987.

5. Pike, Rob, "Why Workstations Don't Work," Summer 1989 USENIX Conference, Baltimore MD.

6. AT&T, "STREAMS Programmer's Guide," UNIX System V Release 3 Manual, Issue 1, Order Code 307-227, 1986.

7. AT&T, "Network Programmer's Guide," UNIX System V Release 3 Manual, Issue 1, Order Code 307-230, 1986.

8. Padovano, Michael, and Scheer, Michael, "UNIX System V Remote File Sharing (RFS)," Summer 1989 USENIX Conference Tutorial Notes, Baltimore MD.

9. Rago, Stephen, "Out-Of-Band Communications in STREAMS," Summer 1989 USENIX Conference Proceedings, Baltimore MD, pp. 29-37.

Elements of Computer Music

By F. Richard Moore

Prentice-Hall (1990) 546 pages
Foreword by Max Mathews

Reviewed by Mike Hawley
MIT Media Lab

In the spectrum of "Elements" books over the centuries, Moore's is rather far from the slim Strunk & White or Kernighan & Plauger end, and pushing decidedly in the direction of Euclid (13 volumes). Computer music is a highly synthetic field – making good music with computers necessarily requires competence in programming (of all kinds, including systems, numerical algorithms, and interfaces), mathematics, signal processing (with an audio bent), perhaps acoustics, probably psychology, and, from time to time, possibly even music. Considering all that, TECM is remarkably concise, yet still touches on a wealth of fine points. Because computing is a profession of mine, and music (including academic computer music) a consuming avocation, I tend to read books of these kinds more for new tidbits than old examples. But the book covers a balance of topics that should resonate well with novices, amateurs, and head-banging die-hard computer music groupies:

Introduction (Musical Data and Process; Musical Thought; Composing; Performing; Instruments; Rooms; Listening; Disciplinary Context; Prerequisites);

Digital Audio (Sound Representations; Sound Digitization; Spectrum Measurements; Digital Filters; Summary);

Instruments (Representing Instruments; cmusic; Additive Synthesis; Subtractive Synthesis and Physical Models; Nonlinear Synthesis; Summary)

Rooms (Concert Halls; Spatial Hearing; Early Echo Response; Reverberation; Sound Spatialization; Other Issues; Summary)

Composing (Computer-Mediated Composition; Music Representations; Random Numbers; Random Sieves; Markov Processes; Noise, Filters, Random Numbers, and Probability; Compositional Algorithms)

Appendices – Mathematics; Units of Measure; Tuning; CMUSIC

Index

Computer music has been attached to (some would say mired in) the domain of signal processing for years, so it's not surprising that the middle three chapters (digital audio, instruments, signal processing) involve this sort of theory and its applications and implementations. In fact, the book actually contains a good bit of the signal processing that every liberal-minded person ought to know and might almost make a workable introductory text for that field. There is a nice perspective on the streams of work done in the field, and extensive examples in the "cmusic" language (a dialect of C with add-ons for music and signal processing). For novices, the fact that so much working example code is presented through C and cmusic is likely to be a boon – so long as the book comes with a diskette, which it apparently does not. Moreover, a book about music without audio illustrations is like an architecture book without pictures of buildings (or a *Kama Sutra* without ...). There seems to be no demo record with this book. It is an expensive undertaking – if not an out-and-out financial loss – to print a computer music book, but an attached recording, particularly if it contained some demos by Prince of sampling synthesis applications or artificial ambience, say, would not only perk up the content, but might even boost the sale if it contained a new song. In fact, although the book does a nice service for the field, I suspect and hope that in five years, if not less, it will be preferable to embody this kind of survey in a computer ware, with running examples. My guess is that the advent of media-rich publishing systems, combined with the recent dramatic shift of attention away from signal processing and more towards content processing (e.g., MIDI), will make this text seem a bit quaint by 2000 A.D.

Readers who would prefer to chaw on the gritty papers that populate the field can survey the *Computer Music Journal* over the years, or perhaps dive into a collection like "Foundations of Computer Music" (Curtis Roads and John Strawn, eds., MIT Press, 1985), which high-pass filters some of the esoterica. Although "great" computer music is tough to find, the field is definitely *not* short on printed matter. These disclaimers aside, though, Moore's book is nevertheless as clear a survey as one could wish for. He is, as Mathews points out in the foreword, not only a technical founder, but a fine teacher and writer. But perhaps better than that, he is also a bona-fide UNIX hacker. Many readers of *;login:* will enjoy tracking some of the familiar systems trails into somewhat new territory.

1989 USENIX Membership Survey

In May 1989 the Executive office created a survey to gather information from the membership to learn more about who you are and what your membership needs are. The survey was mailed to 1,325 individual members. We received 498 responses – a whopping 37% rate of return!

Your feedback is useful to us. You want more technical articles in *;login:*, more articles in *Computing Systems*, you raved about the C++ and Systems Administration Workshops; and you want more focus on networking.

One of our charters in 1990 is to turn your recommendations into a reality. We appreciate your responses and look forward to a continuing dialogue.

Here is a more detailed review of your answers to the open-ended questions and a graphic overview of the entire survey.

What changes would you like to see in *;login:*?

More technical material
Articles on trends in the industry
A tips and hints/question and answer column

What changes would you like to see in *Computing Systems*?

A volume devoted to a particular topic
Articles that elaborate on the papers from conferences
More articles that include graphics/illustrations
More how/to applied articles
More issues!

Of the technical conferences you attended, which were the best?

Of the 89 who answered this question:

34 for San Francisco
15 for Dallas
9 each for Phoenix, San Diego, Portland and Atlanta
Balance spread among Toronto, Salt Lake City, Denver and D.C.

Of the workshops you attended which were the best?

Overwhelmingly: C++ and Systems Adminstration

Please list any topics you consider suitable for future USENIX workshops.

There were many responses. The most frequestly mentioned were:

Networking concerns – management and security
OSI and UNIX
Standards

X Windows
Distributed/Multiprocessor Systems
Mach

Please list any projects (like UUNET or Face Saver) in which the Association might invest. (Some of these have been accomplished by USENIX or UUNET.)

Most of the respondents were enthusiastic about UUNET and Face Saver, but had very few new ideas to offer. Here are a few:

Have all the USENIX papers online
Tutorials by mail/video
Software portability guidelines
An anonymous ftp site for USENIX sources
Regional tutorials
Funding for a complete rewrite of the news software
An information retrieval system which could access contents of UUNET, News, FTP archives, publicly funded R&D efforts, etc.
Literature database related to the computing industry – focusing on UNIX-related subjects
Portability checker for C source code and a library of portable, reusable software libraries and programs
More research in efficient news distribution
A central site for public domain software in source form

We would appreciate any suggestions you might have that can be incorporated into future surveys. Please contact the Executive Director (*ellie@usenix.org*).

–EY

1989 USENIX Membership Survey

USENIX Information

Do you read *;login:*? Always----------Sometimes----------Never

If you do, please remark on its content:

	Worthless	Average	Excellent
Technical content (articles, reviews)	0----------1	----------2	----------3----------4
Information regarding workshops & conferences	0----------1	----------2	----------3----------4
Information regarding user groups	0----------1	----------2	----------3----------4

What changes would you like to see in *;login:*?

Do you read *Computing Systems* ? Always----------Sometimes----------Never

If you do, please remark on its content:

	Worthless	Average	Excellent
Technical content (articles, reviews)	0----------1	----------2	----------3----------4
Format	0----------1	----------2	----------3----------4

What changes would you like to see in *Computing Systems* ?

In 1987, the Association distributed two tapes containing 70MB of software derived from net.sources and mod.sources; in 1988 a tape of C++ software was produced. Should the Association continue to produce and distribute software tapes? Yes----------No

Personal Information

Are you employed by:
- ❑ a business or corporation
- ❑ an educational institution
- ❑ a government agency
- ❑ a nonprofit organization (other than educational)
- ❑ a research facility

or are you:
- ❑ self–employed
- ❑ a full–time student
- ❑ other _____

Approximately how many are in your organization? _____

Are you:
- ❑ an executive or senior manager
- ❑ a dean or department chair
- ❑ a project leader or senior programmer
- ❑ a professor or instructor
- ❑ a programmer or technical staff member
- ❑ a consultant or self–employed
- ❑ a student
- ❑ other _____

Have you completed:
- ❑ a doctorate
- ❑ a masters
- ❑ some graduate work
- ❑ a baccalaureate
- ❑ college without a degree
- ❑ high school

How many years have you been a USENIX member? ____

Are you a member of:
- ❑ ACM
- ❑ IEEE
- ❑ /usr/group
- ❑ other professional organizations _____ _____

How many years have you been using UNIX? ____

How old are you? _____

Are you ❑ male or ❑ female

What is your annual salary or income?

under $10K------$10K------$20K------$30K------$50K------$70K------$100K------over $100K

Conferences and Workshops Information

If you have attended any USENIX conference or workshop in the past three years, please give your general assessment:

	Unsatisfactory	Fair	Good
❑ Denver Technical Conference 1986	☐	☐	☐
❑ Atlanta Technical Conference 1986	☐	☐	☐
❑ Monterey Graphics Workshop 1986	☐	☐	☐
❑ Washington Technical Conference 1987	☐	☐	☐
❑ Philadelphia Systems Administrators' Workshop 1987	☐	☐	☐
❑ Phoenix Technical Conference 1987	☐	☐	☐
❑ Cambridge Graphics Workshop 1987	☐	☐	☐
❑ Berkeley POSIX Workshop 1987	☐	☐	☐
❑ Santa Fe C++ Workshop 1987	☐	☐	☐
❑ Dallas Technical Conference 1988	☐	☐	☐
❑ IEEE/USENIX Real–Time Workshop 1988	☐	☐	☐
❑ San Francisco Technical Conference 1988	☐	☐	☐
❑ Portland Security Workshop 1988	☐	☐	☐
❑ Pittsburgh Supercomputing Workshop 1988	☐	☐	☐
❑ Denver C++ Conference 1988	☐	☐	☐
❑ Monterey Systems Administrators' Workshop 1988	☐	☐	☐
❑ San Diego Technical Conference 1988	☐	☐	☐

Of the technical conferences you attended, which were the best?

Of the workshops you attended, which were the best?

Please list any topics you consider suitable for future USENIX workshops.

Do you prefer concurrent Winter conferences with /usr/group? Yes----------No---------Don't Care

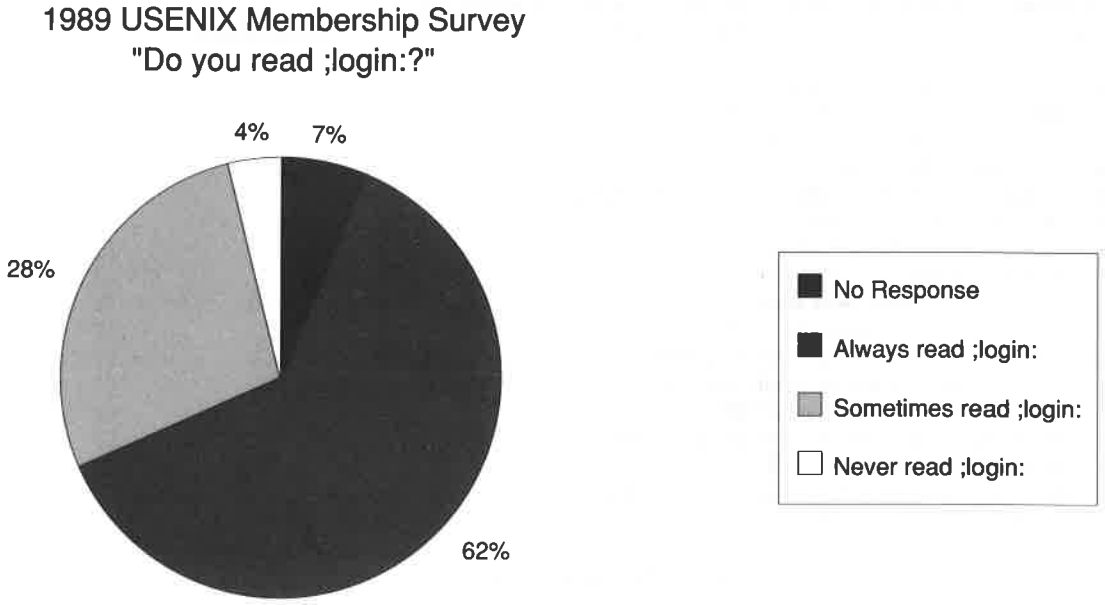
Please list any projects (like UUNET or Face Saver) in which the Association might wisely invest.

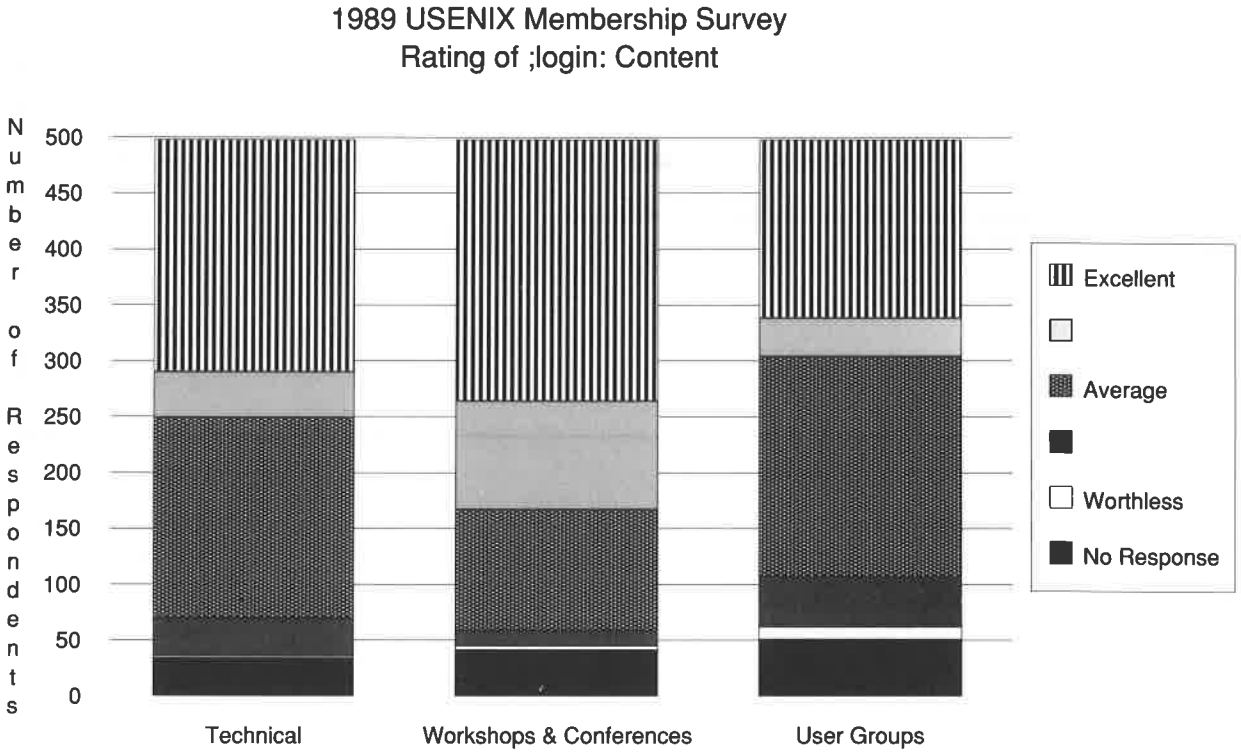
Thank you for completing this year's survey.

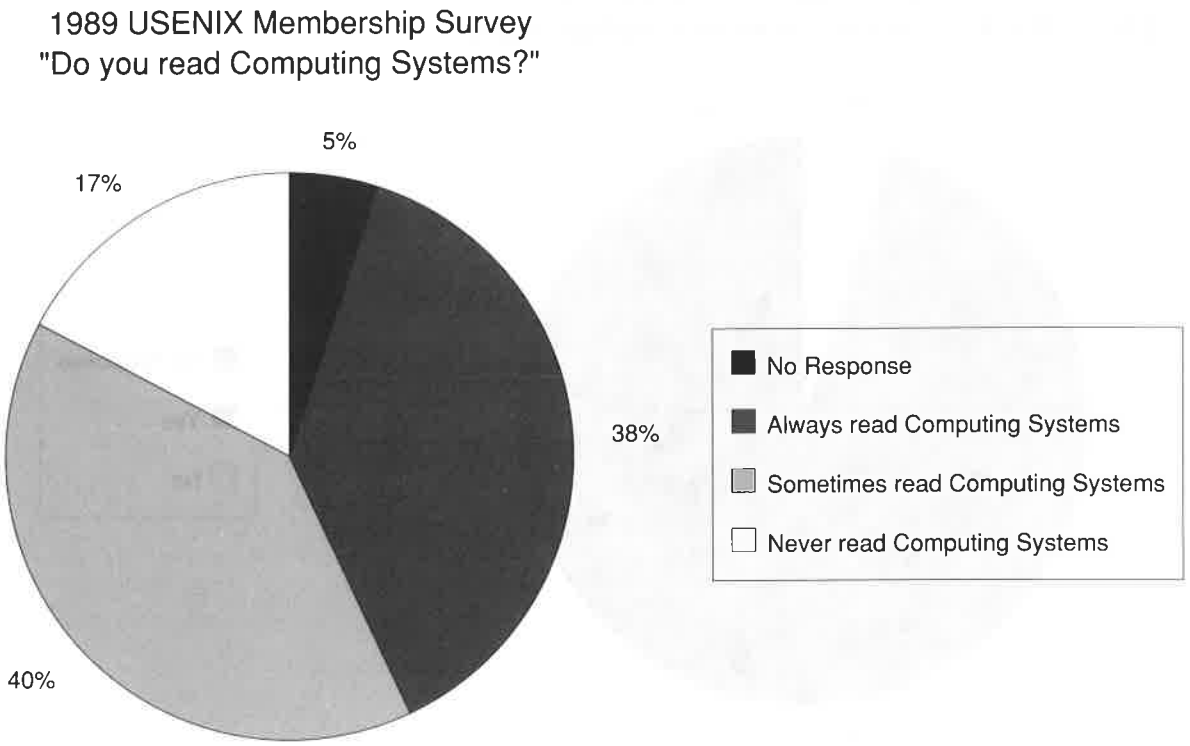
Your (nosey) Association staff

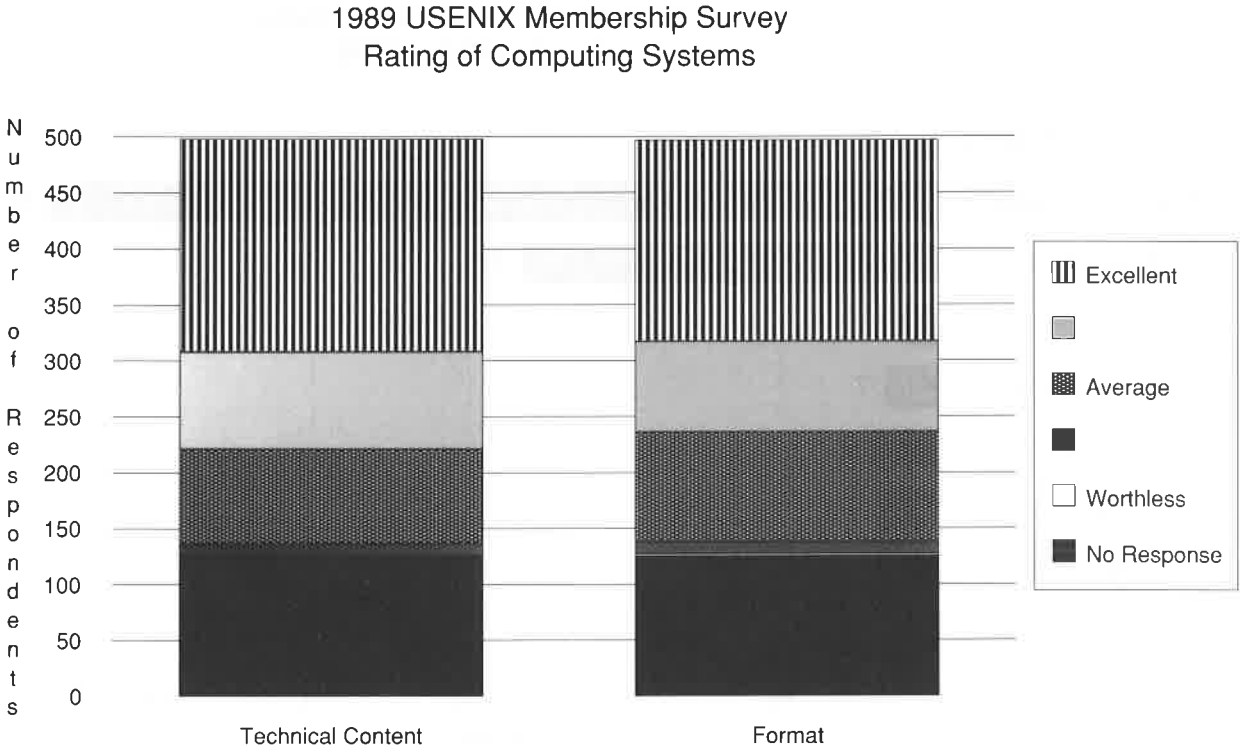
Please return this survey to:

USENIX Association
2560 Ninth Street
Berkeley, CA 94710

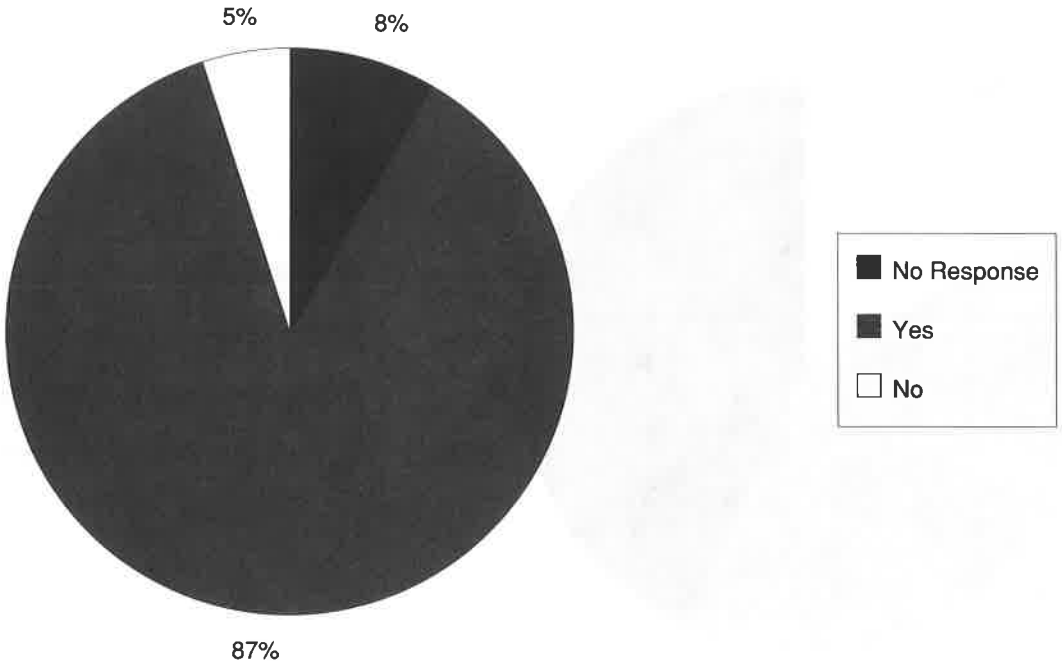

1989 USENIX Membership Survey
"Do you read ;login:?"
4%
7%
28%
62%
No Response
Always read ;login:
Sometimes read ;login:
Never read ;login:


1989 USENIX Membership Survey
Rating of ;login: Content
Number of Respondents
500
450
400
350
300
250
200
150
100
50
0
Technical
Workshops & Conferences
User Groups
Excellent
Average
Worthless
No Response

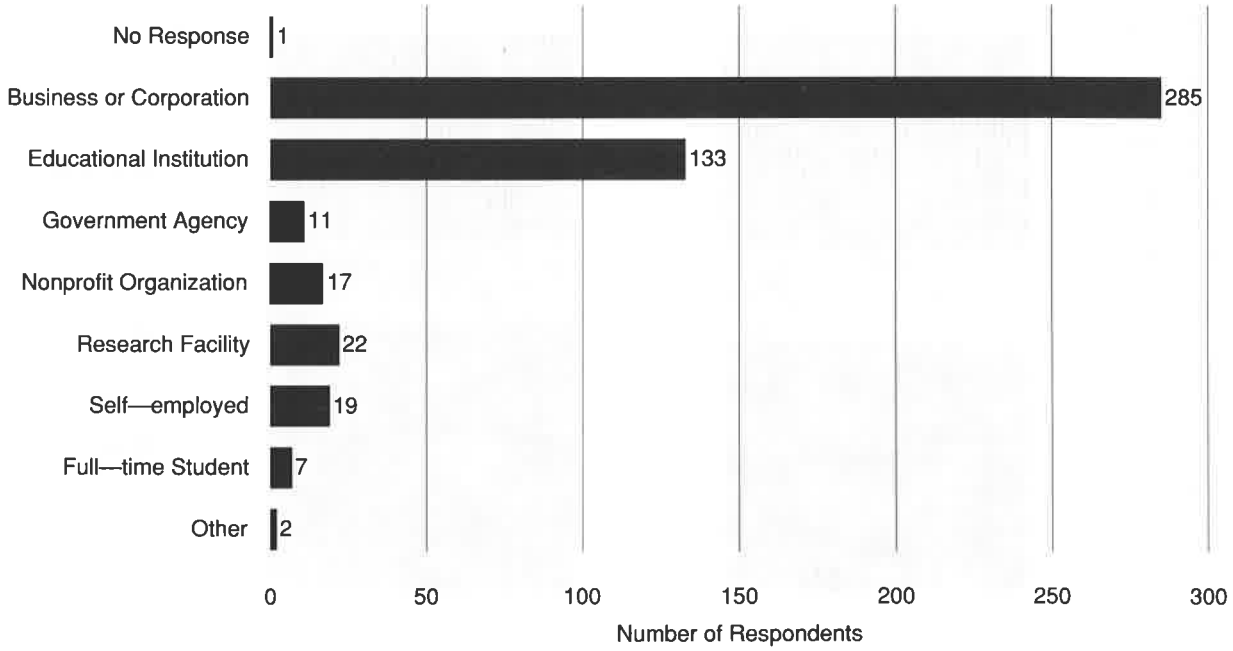

1989 USENIX Membership Survey
"Do you read Computing Systems?"
5%
17%
38%
40%
No Response
Always read Computing Systems
Sometimes read Computing Systems
Never read Computing Systems

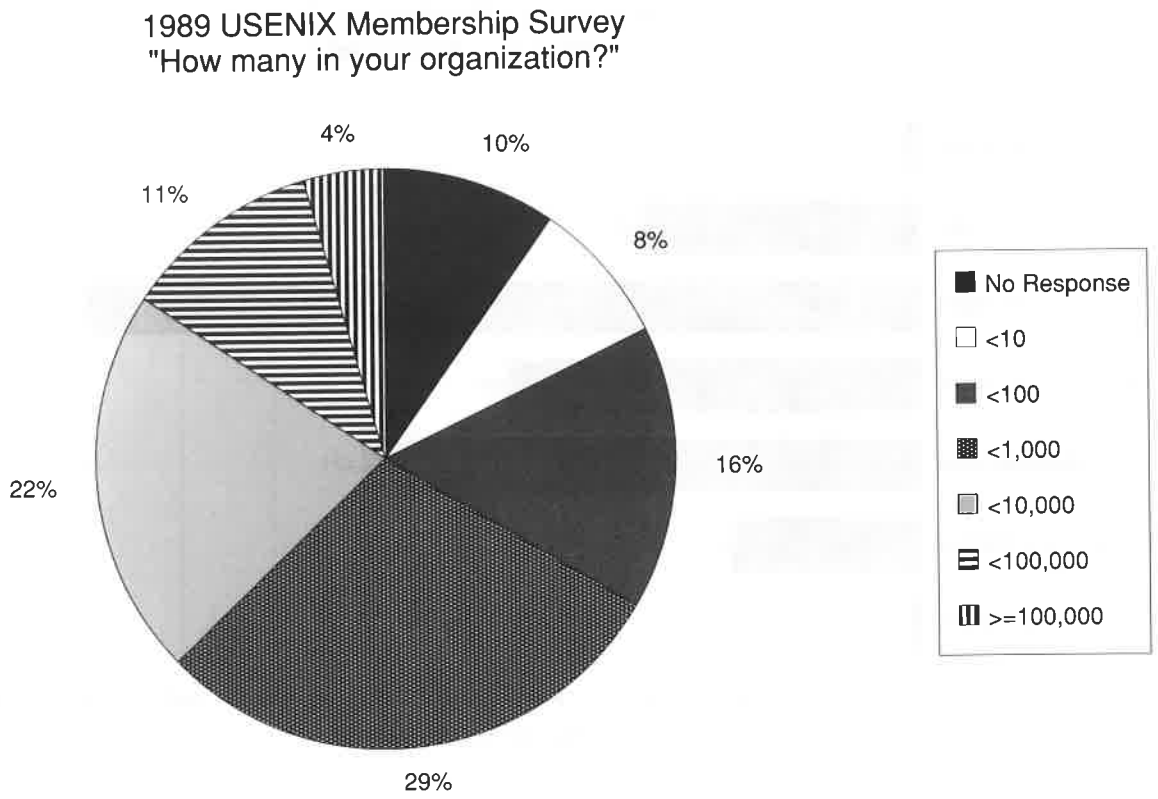

1989 USENIX Membership Survey
Rating of Computing Systems
Number of Respondents
500
450
400
350
300
250
200
150
100
50
0
Technical Content
Format
Excellent
Average
Worthless
No Response

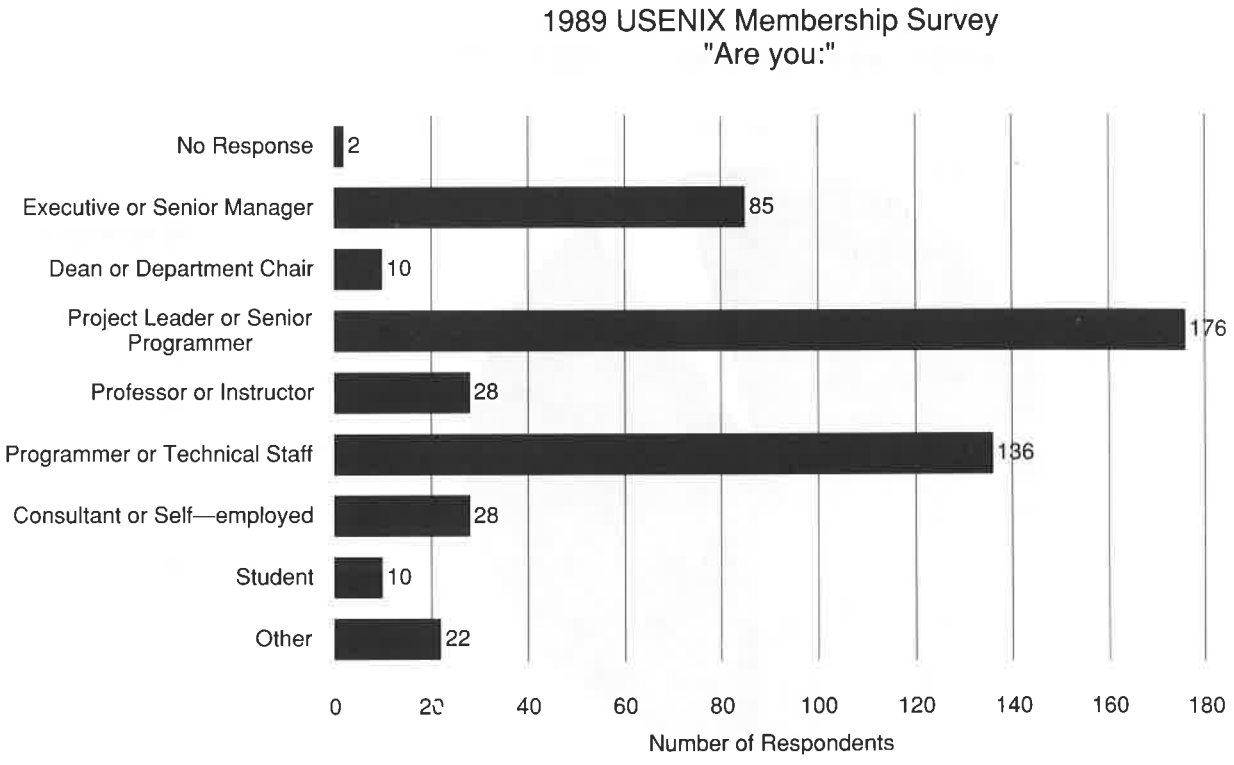
1989 USENIX Membership Survey
"Should the Association continue to produce tapes?"

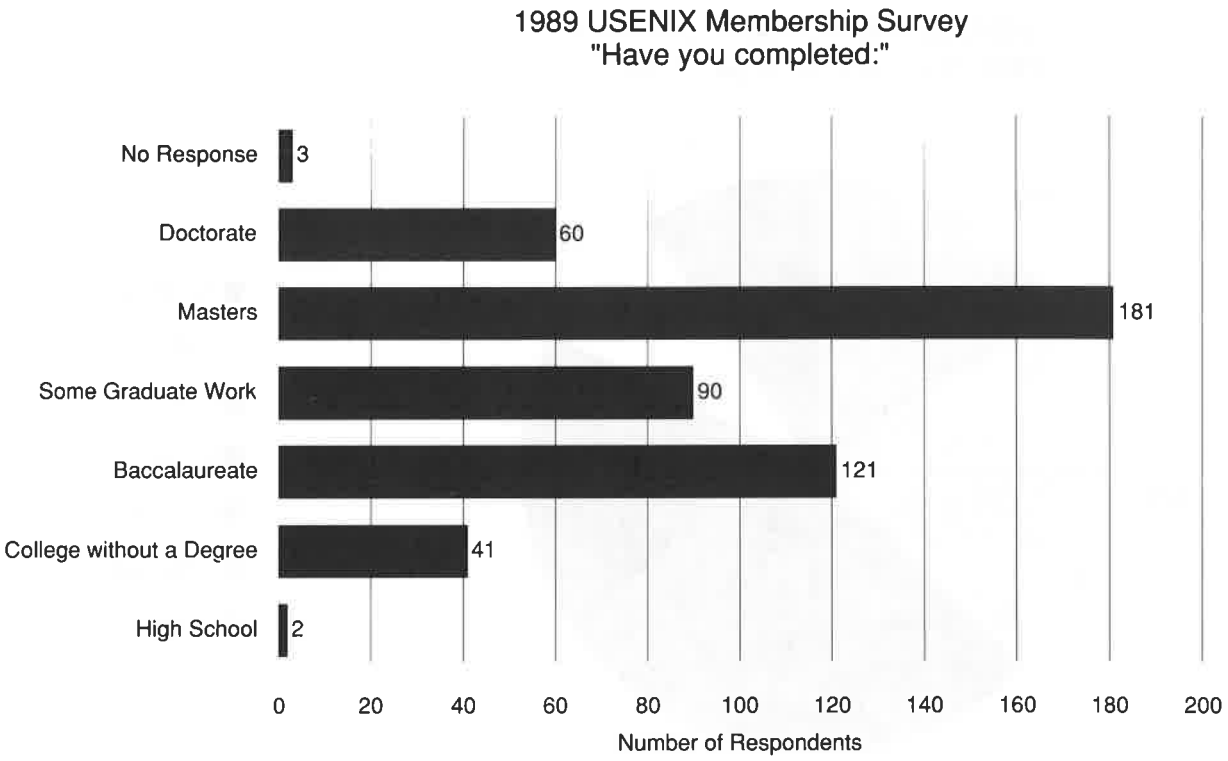


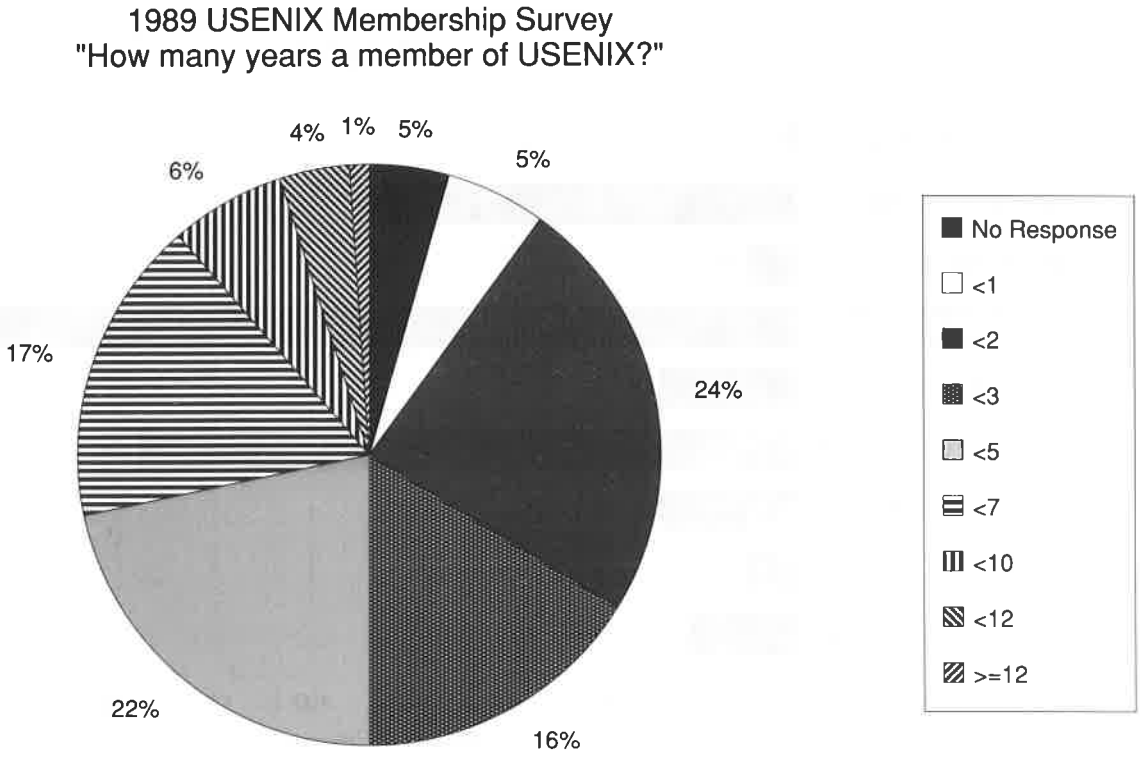
1989 USENIX Membership Survey
"Are you employed by:"

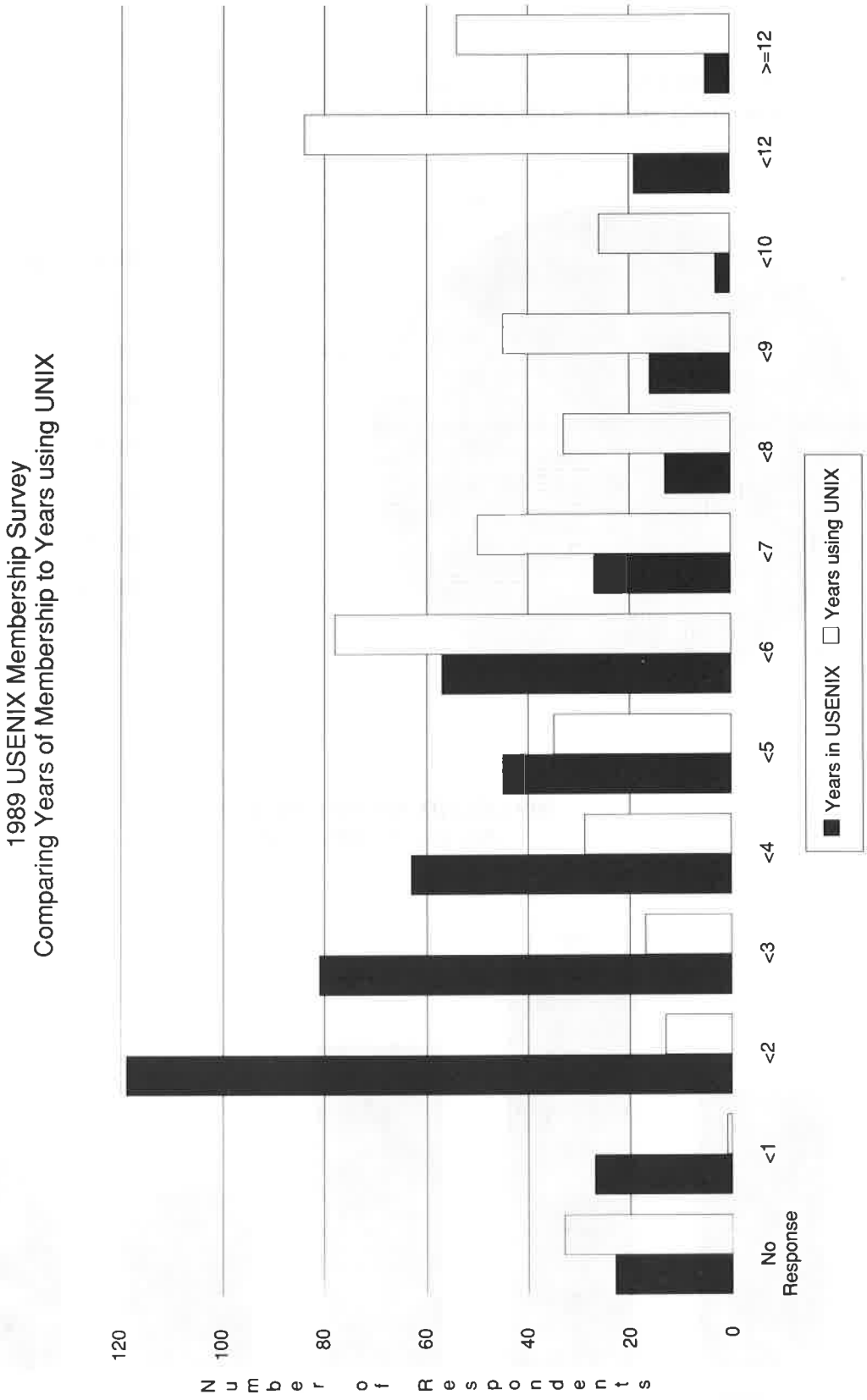


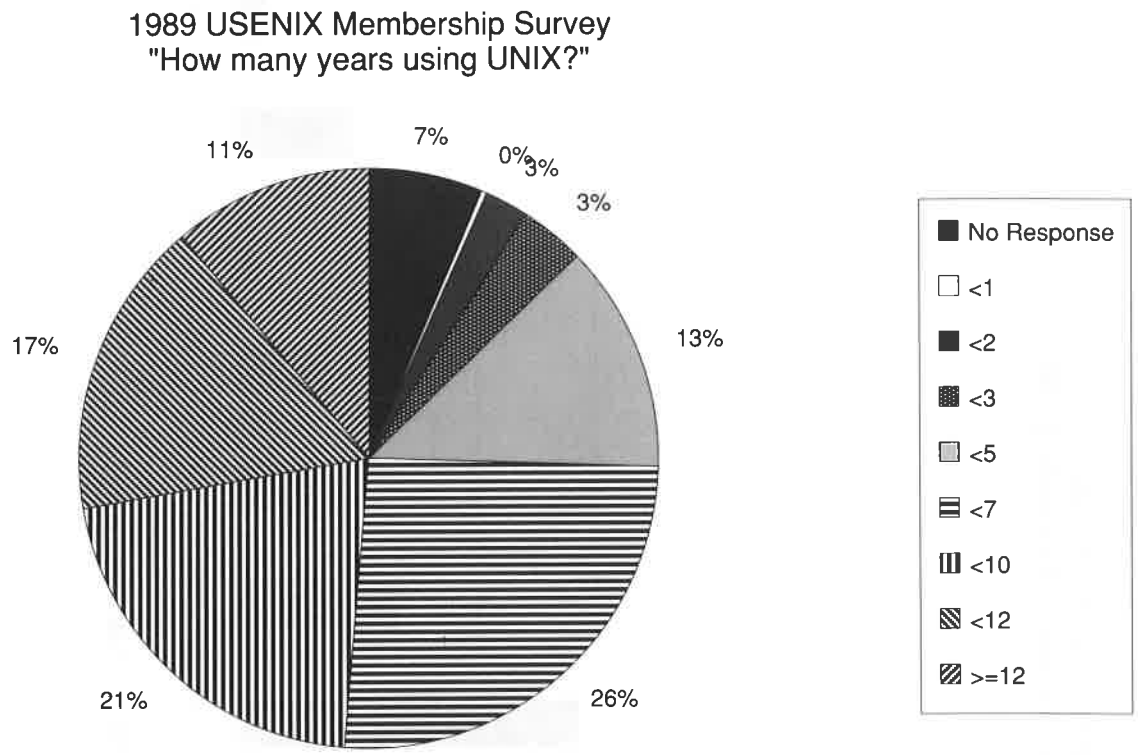

1989 USENIX Membership Survey
"How many in your organization?"
4%
10%
11%
8%
16%
22%
29%
No Response
<10
<100
<1,000
<10,000
<100,000
>=100,000

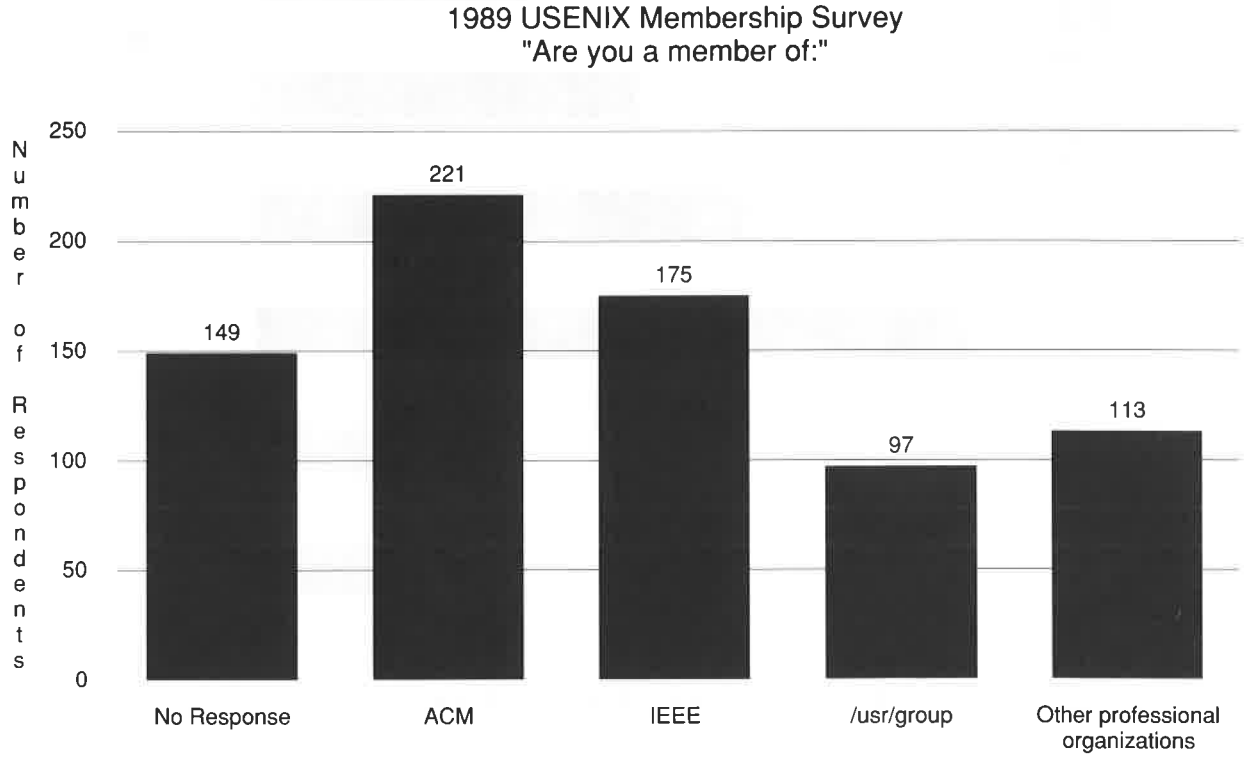

1989 USENIX Membership Survey
"Are you:"
No Response
2
Executive or Senior Manager
85
Dean or Department Chair
10
Project Leader or Senior Programmer
176
Professor or Instructor
28
Programmer or Technical Staff
136
Consultant or Self—employed
28
Student
10
Other
22
0
20
40
60
80
100
120
140
160
180
Number of Respondents

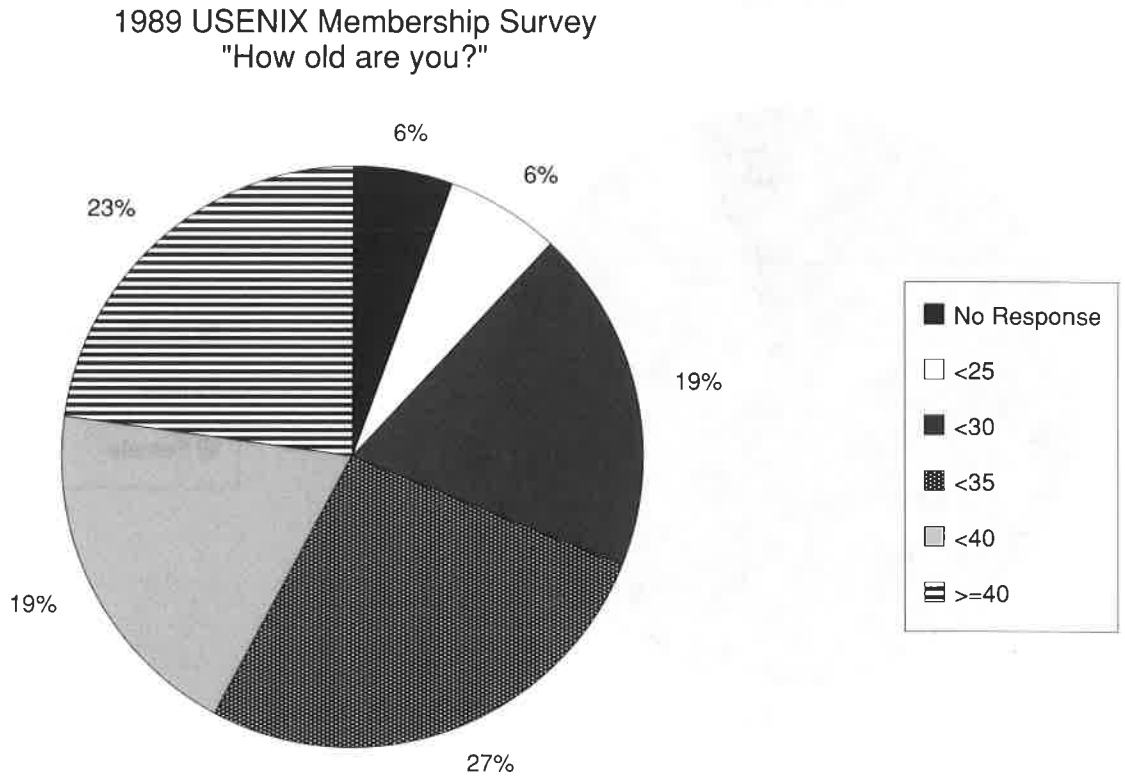

1989 USENIX Membership Survey
"Have you completed:"
No Response
3
Doctorate
60
Masters
181
Some Graduate Work
90
Baccalaureate
121
College without a Degree
41
High School
2
0
20
40
60
80
100
120
140
160
180
200
Number of Respondents

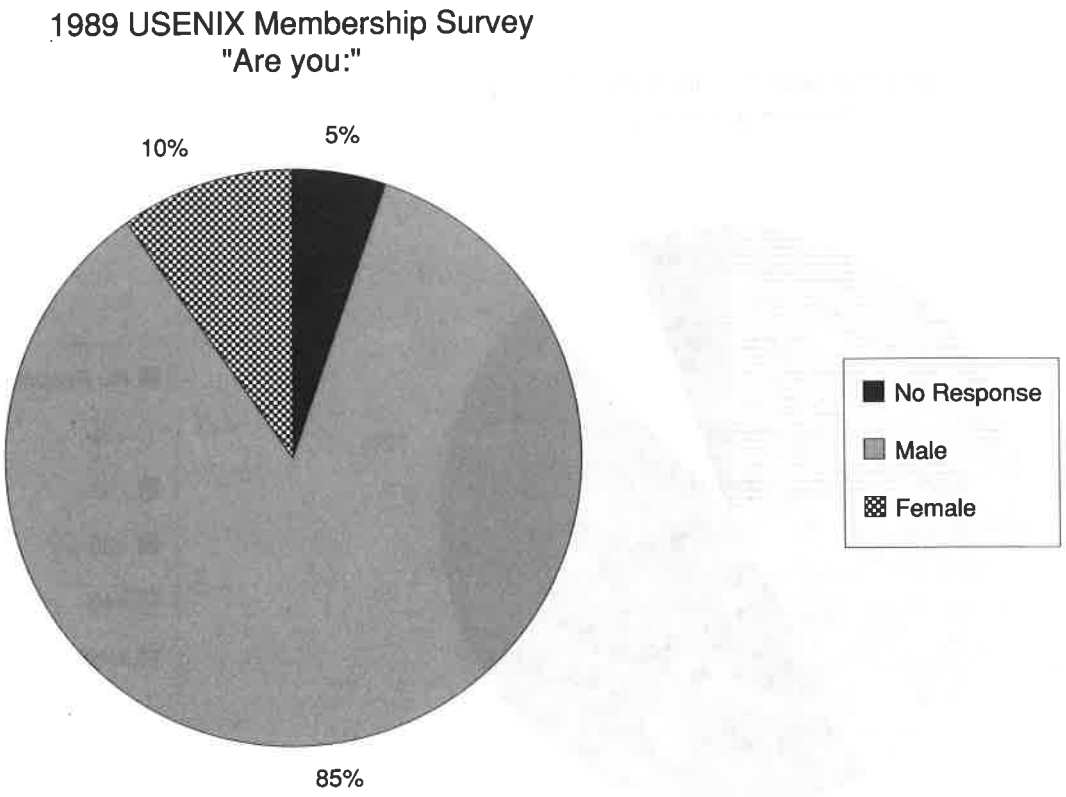

1989 USENIX Membership Survey
"How many years a member of USENIX?"
4%
1%
5%
5%
6%
17%
24%
22%
16%
No Response
<1
<2
<3
<5
<7
<10
<12
>=12

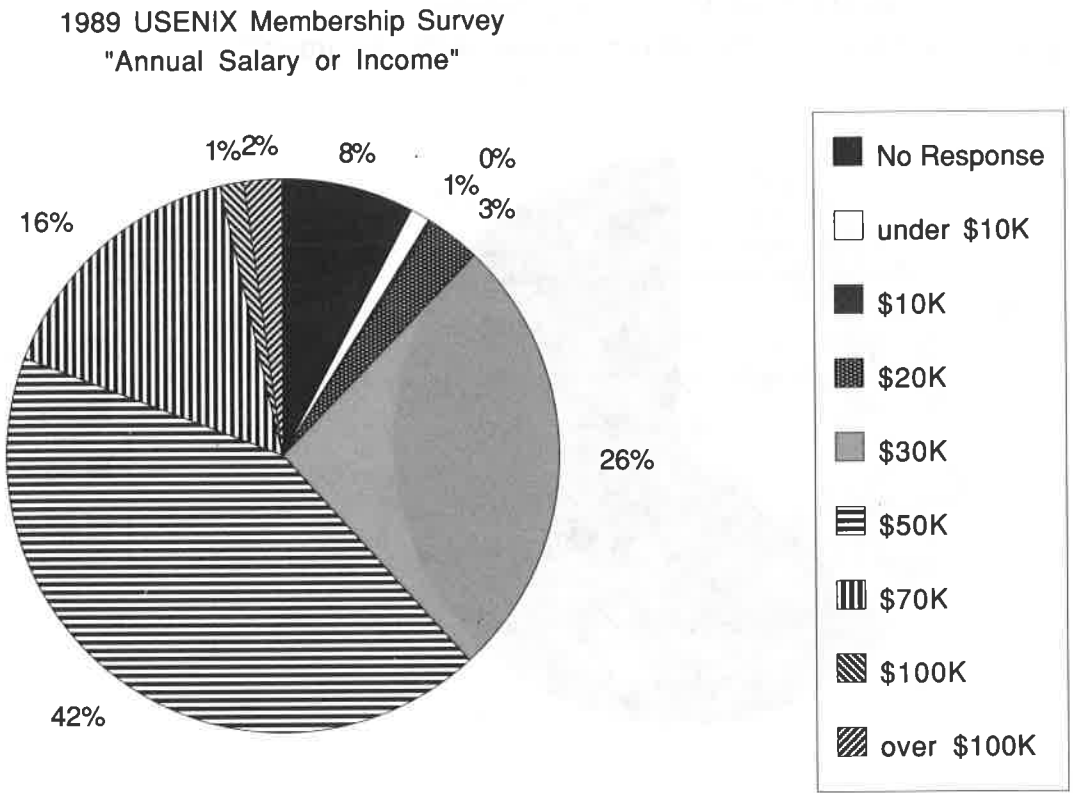

1989 USENIX Membership Survey
Comparing Years of Membership to Years using UNIX
Number of Respondents
120
100
80
60
40
20
0
No Response
<1
<2
<3
<4
<5
<6
<7
<8
<9
<10
<12
>=12
Years in USENIX
Years using UNIX


1989 USENIX Membership Survey
"How many years using UNIX?"
7%
0%
3%
3%
13%
26%
21%
17%
11%
No Response
<1
<2
<3
<5
<7
<10
<12
>=12

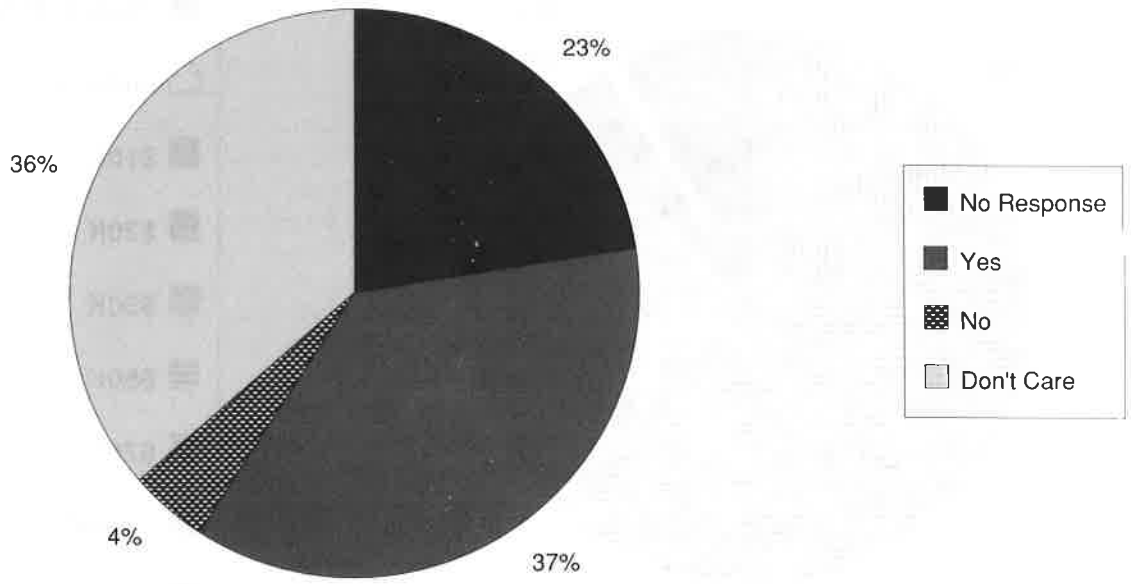

1989 USENIX Membership Survey
"Are you a member of:"
Number of Respondents
250
200
150
100
50
0
149
221
175
97
113
No Response
ACM
IEEE
/usr/group
Other professional organizations

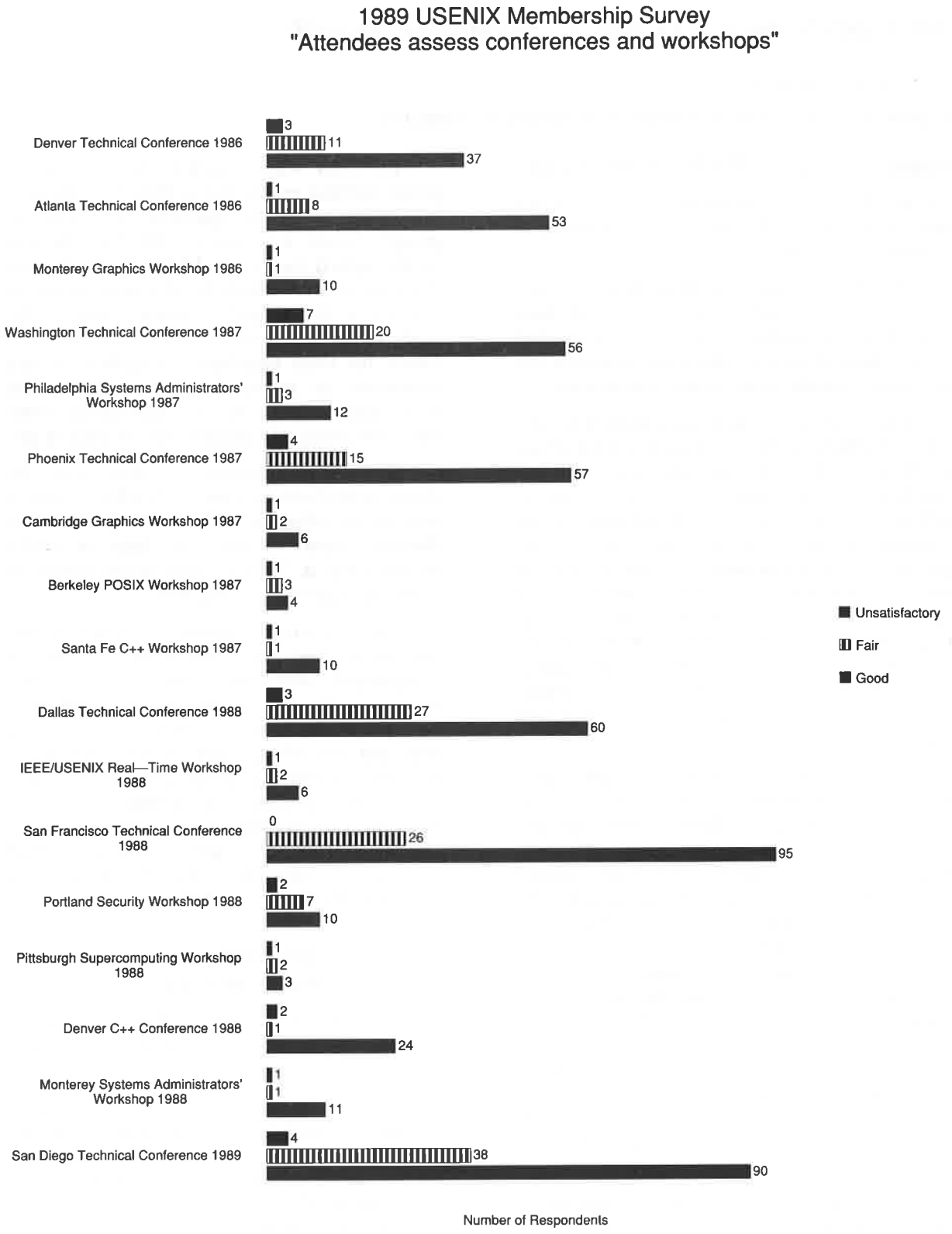

1989 USENIX Membership Survey
"How old are you?"
6%
6%
23%
19%
19%
27%
No Response
<25
<30
<35
<40
>=40


1989 USENIX Membership Survey
"Are you:"
10%
5%
85%
No Response
Male
Female


1989 USENIX Membership Survey
"Annual Salary or Income"
1%
2%
8%
0%
1%
3%
16%
26%
42%
No Response
under $10K
$10K
$20K
$30K
$50K
$70K
$100K
over $100K

1989 USENIX Membership Survey
"Do you prefer concurrent Winter conferences with /usr/group?"




1989 USENIX Membership Survey
"Attendees assess conferences and workshops"
Denver Technical Conference 1986
3
11
37
Atlanta Technical Conference 1986
1
8
53
Monterey Graphics Workshop 1986
1
1
10
Washington Technical Conference 1987
7
20
56
Philadelphia Systems Administrators' Workshop 1987
1
3
12
Phoenix Technical Conference 1987
4
15
57
Cambridge Graphics Workshop 1987
1
2
6
Berkeley POSIX Workshop 1987
1
3
4
Santa Fe C++ Workshop 1987
1
1
10
Dallas Technical Conference 1988
3
27
60
IEEE/USENIX Real—Time Workshop 1988
1
2
6
San Francisco Technical Conference 1988
0
26
95
Portland Security Workshop 1988
2
7
10
Pittsburgh Supercomputing Workshop 1988
1
2
3
Denver C++ Conference 1988
2
1
24
Monterey Systems Administrators' Workshop 1988
1
1
11
San Diego Technical Conference 1989
4
38
90
Unsatisfactory
Fair
Good
Number of Respondents

An Update on UNIX and C Standards Activity

Jeffrey S. Haemer
Report Editor, USENIX Standards Watchdog Committee

Report on IEEE 1003.0: POSIX Guide

Kevin Lewis <klewis@gucci.enet.dec.com> reports on the October 16-20, 1989 meeting in Brussels, Belgium:

Dot Zero's mission in Brussels was to step back and review where the group had been and where we needed to go. We learned that we are headed in the right basic direction but still need to make some course corrections.

There are two major contributors to this state of affairs. First, an honest review of the pre-Brussels document reveals that it still has significant holes. Also, its format is hard to follow. I must admit that it felt good to see unanimous consent on both the need to reorganize the document and on a new format. It does a co-chair's heart good to see two such rare events occur concurrently. The reformatting of the draft guide will be complete by the January meeting in New Orleans. The group will then review components of the document that are sufficiently complete section-by-section and line-by-line.

Second, Dot Zero faces a problem that is becoming widespread in 1003 and TCOS-SS: a serious dilution of effort. Little did Dot Zero realize, when it recommended the formation of a group to address a windows standard (now 1201), that we would lose people who had been shepherding key components of the Dot Zero guide. The new efforts have left us with no one to cover networking, graphics, or windows, though it's possible that new folks in these areas will join us in New Orleans. [Editor's note: Listen to this man. What are your ideas about open systems in these areas? If you have something useful to contribute, please contact someone on Dot Zero : Kevin. Don't wait until it's too late and then complain about the result.]

Regarding internationalization (for which the current buzzword is "I18N," because there are eighteen letters between the 'I' and the 'N'):

Everyone who attended the I18N study group meeting sponsored by Dot Zero found it interesting when the question regarding the group's future was posed. All those present tacitly agreed that it would not be in the best interests of I18N efforts for this study group to become a full-fledged working group. This study group would best serve the industry as a forum for issue flagellation, soap-boxing, and formation of proposals to the appropriate accredited bodies. At the appropriate time, the I18N group will declare that its time is up.

When the question of identifying the major contributors to the I18N efforts arose, I noticed an effort of OSF to remain at arm's distance from X/Open. In light of OSF's membership in X/Open, this might signify its desire to maintain its own identity.

That's enough negatives. Is there an upside to all this? Yes, absolutely. We have a reorganized document that will ease and streamline the review process. We now have the eyes of the industry and the press looking over our shoulders, eager to read our guide. We are reaching the point where fear of personal and professional embarrassment is motivating those who have an interest in this effort's succeeding. These will help us meet our goal of preparing a draft for review and comment by ISO by the Fall of 1990.

Report on IEEE 1003.1: System Services Interface

Mark Doran <md@inset.co.uk> reports on the October 16-20, 1989 meeting in Brussels, Belgium:

P1003.1 is now a full-use standard, so interest in attending the working group has waned somewhat. Attendance didn't get above fifteen or so all week and was nearer half a dozen most of the time. Even so, this was a bit low by comparison with recent meetings. So where was everyone?

[Editor's note – Notice that this is larger than the attendance at typical meetings of, for example, dot nine. "Low attendance" is relative. Author's additional note – And that's the frightening thing; standards are being established by as few as half a dozen *individuals.* This cannot be representative or balanced. Scary stuff, "...as we take you on a journey, into the Standards Zone..."]

We thought that meeting in Brussels was going to further the cause of international participation in the POSIX process. Several people I would normally expect to see, didn't show; Europe may be too far from home for a lot of the regulars. Unfortunately, I didn't see more than two or three Europeans (whom I would not normally expect to see at POSIX) all week either. Oh well, I'm sure it was a good idea really... So what did those that showed get up to?

ISO 9945. [Editor's note – ISO 9945 is, roughly, the ISO standard engendered by and corresponding to the POSIX work.]

It looks like 9945 is going to be split up into three major pieces, the first of which is founded upon the IEEE P1003.1-1988 standard. This piece is likely to include all the other system interfaces as well (notably, the real time stuff from P1003.4). The other two pieces will be based around utilities and system administration.

The basic IS9945-1:1989 will be just the same as the regular, ugly-green, dot-one book – well almost. ISO has yet another documentation format, and the book will have to be squeezed to fit it. This one doesn't allow line numbers either. We are assured that making the changes is not a major problem, but the working group has still requested a new disclaimer telling readers that all mistakes are the fault of ISO!

P1003.1a. [Editor's note – This document (supplement A) is supposed to contain clarifications of and corrections to P1003.1-1988, but no substantive changes.]

The meeting discussed resolution issues from the first ballot. Highlights included:

- the decision to withdraw the *cuserid()* interface; its loss will not be sadly mourned since one can use other interfaces to do the same job better.

- the addition of a new type name `ssize_t` to represent signed `size_t` values; this has all sorts of uses – for example, in the prototype for *read()*. Currently, the parameter specifying the number of bytes to be *read()* is given as a `size_t`, but *read()* has been specified to return an `int`, which may not be large enough to hold a `size_t` character count. Moreover, *read()* may return -1 for failure, or the number of characters read if the call was successful.

The recirculation ballot happened between November 10-20, 1989; if you care but didn't know that already, it doesn't matter because you (and many others, I suspect) have missed your chance. This all seems a bit fast but it does mean that P1003.1a will hit an ISO, six-month, letter-ballot window.

Transparent File Access. Isn't this a P1003.8 problem? Yes, but the chairperson of the TFA committee came to talk about TFA semantics as they relate to P1003.1.

The crux of the matter is that the six TFA folks seem to have decided that standardizing NFS will do nicely for TFA. Their chairperson wonders whether the rest of the world (or, more accurately, the balloting group for a TFA standard) will agree.

The answer from the dot one folks appears to be definitely not (thank goodness)! There appear to be several arguments against NFS as the TFA standard from dot one. These include:

- It is impossible to maintain full dot one semantics over a network using NFS. Consider the last-close semantics, for example, which can only be preserved over a network using a connection-oriented protocol, which NFS is not.

- Transparent File Access should be *transparent;* NFS isn't. It is possible for operations that are logically sound to fail because of network timeouts.

- NFS is a *de facto* standard; why should it get an official rubber stamp?

This appears to be a hot topic that many groups may have an interest in, so there will

be an "out-of-hours" meeting on TFA at the New Orleans POSIX – [Editor's note – If you do care, we suggest either writing directly to the TFA chair, Jason Zions <jason@hpcndr.cnd.hp.com>, or posting your opinions to comp.std.unix.]

Language-Independent Specification. It seems to have been decided that POSIX API standards should be written in a language-independent form, i.e. not expressed in C-language constructs.

My initial reaction was one of horror, but then someone pointed out that C is not the only programming language in the known universe. This I have to concede, along with the idea that bindings to POSIX APIs in other languages may not be such a bad idea after all. Indeed work is well underway to produce FORTRAN and Ada bindings.

But now it seems we have to express POSIX in a language-independent way. When you come to write the next set of actual language bindings, the semantics won't be clouded with language-dependent stuff; the idea is that you won't have to understand C in all its "glory" to write new language bindings.

So what will the language-independent specifications look like? Will I be able to understand those? The current proposal doesn't look like anything I recognize at all. Yes, that's right, we have to learn a whole NEW language (sigh). Why not use a more formal specification language that a few people know? (Like ASN.1 for example, which P1003.7 has decided to use.) Better yet, why not use constrained English? Since the FORTRAN and Ada bindings folks have managed without the aid of language-independent specifications, why can't everyone else? Is there more to this than a glorified job creation scheme? ("Wanted: expert in POSIX 'language-independent' language...") If there is, do we have to invent a new wheel to get the job done?

As you can tell, my opinion of this effort is somewhat jaundiced. Perhaps, I have missed the point. Maybe so, but if I have, I feel sure that some kind soul will be only too happy to correct me in "flaming" detail.

Messaging. The UniForum internationalization (I18N) folks brought forward a proposal for a messaging facility to be included in P1003.1b. The working group decided that it needs some more work but will go into the next draft.

[Editor's note – The problem being solved here is that internationalized applications store all user-visible strings in external files, so that vendors and users can change the language of an application without recompiling it. The UniForum I18N group is proposing a standard format for those files.]

P1003.1b. Work on production of the second supplement is still at a formative stage. The group is still accepting formal proposals for functionality to add to the document. Where P1003.1a has been drawn up as a purely corrective instrument, P1003.1b may add new functionality. Among the interesting things currently included are these:

- The messaging proposal described above.
- A set of interfaces to provide symbolic links. The basic idea is that *lstat()*, *readlink()* and *symlink()* operate on the link, and all other interfaces operate on the linked-to file.

Rationale will be added to explain that it is a unique directory, which is the parent directory in the same physical file system. This means that *cd* does not go back across symlinks to the directory you came from.

This is the same as the semantics on my Sun. For example:

```
(sunset) 33 % pwd
/usr/spare/ins.MC68020/md/train
(sunset) 34 % ls -ld ./MR_C++
lrwxrwxrwx  1 root  32 Sep 30  1988 MR_C++
   -> /usr/sunset/md/c++/trainset/c++/
(sunset) 35 % cd MR_C++
(sunset) 36 % pwd
/usr/sunset/md/c++/trainset/c++
(sunset) 37 % cd ..
(sunset) 38 % pwd
/usr/sunset/md/c++/trainset
```

The rationale is meant to help keep readers aware of what's really written in the standard and help them avoid misinterpreting it along lines of their own potential misconceptions.

• P1003.1b used to have two descriptions of Data Interchange formats. Now it has only one. The working group picked the one that remains because it more closely resembles existing standards in the area. In particular, the surviving proposal refers to X3.27.

Report on IEEE 1003.4: Real-time Extensions

John Gertwagen <jag@laidbak> reports on the November 13-17, 1989 meeting in Milpitas, CA:

Background

The P1003.4 Real-time Working Group, began as the /usr/group technical committee on real-time extensions. About two years ago, it was chartered by the IEEE to develop minimum extensions to POSIX to support real-time applications. Over time its scope has expanded, and P1003.4 is now more a set of general interfaces that extend P1003.1 than a specifically real-time standard. Its current work is intended to support not only real-time, but also database, transaction processing, Ada runtime, and networking environments. The group is trying to stay consistent with both the rest of POSIX and other common practice outside the real-time domain.

The work is moving quickly. Though we have only been working for two years, we are now on Draft 9 of the proposed standard, and expect to go out for ballot before the end of the year. To help keep up this aggressive schedule, P1003.4 made only a token appearance at the official P1003 meeting in Brussels. The goal of the Milpitas meeting was to get the draft standard ready for balloting.

Meeting Summary

Most of the interface proposals are now relatively mature, so there was a lot of i-dotting and t-crossing, and (fortunately) little substantive change. The "performance metrics" sections of several interface chapters still need attention, but there has been little initiative in the group to address them, so it looks like the issues will get resolved during balloting.

The biggest piece of substantive work was a failed attempt to make the recently introduced threads proposal clean enough to get into the ballot. The stumbling block is a controversy over how to deal with signals.

There are really two, related problems: how to send traditional UNIX/POSIX signals to a multi-threaded process, and how to asynchronously interrupt a thread.

Four options have been suggested: delivering signals to a (somehow) privileged thread, per Draft 8; delivering a signal to whichever thread is unlucky enough to run next, a la Mach; delivering the signal to each thread that declares an interest; and ducking the issue by leaving signal semantics undefined.

We haven't been able to decide among the options yet; the working group is now split evenly. About half think signal semantics should follow the principle of least surprise, and be fully extended to threads. The other half think each signal should be delivered to a single thread, and there should be a separate, explicit mechanism to let threads communicate with one another.

Personally, I think the full semantics of process signals is extra baggage in the "lightweight" context of threads. I favor delivering signals to a privileged thread – either the "first" thread or a designated "agent" – and providing a separate, lightweight interface for asynchronously interrupting threads. On the other hand, I would be happy to see threads signal one another in a way that looks, syntactically and semantically, like inter-process signals. In fact, I think the early, simpler versions of *signal()* look a lot like what's needed (around V6 or so). I don't care whether the implementation of "process" and "thread" signals is the same underneath or not. That decision should be left to the vendor.

Directions

Draft 9 of P1003.4 is being readied for ballot as this is being written and should be distributed by mid-December. With a little luck, balloting will be over by the Summer of '90.

Threads is the biggest area of interest in continued work. The threads chapter will be an appendix to Draft 9 and the balloting group will be asked to comment on the threads proposal as if it were being balloted. Unless there is a significant write-in effort, the threads interface will probably be treated as a new work item for P1003.4. Then, if the outstanding issues can be resolved expediently, threads could go to ballot as early as April '90.

With the real-time interfaces defined, it looks like the next task of this group will be to create one or more Real-time Application Portability Profiles (RAPPs?) that define how to use the interfaces in important real-time application models. Agreeing on the application models may be harder than agreeing on the interfaces, but we'll see.

Report on IEEE 1003.6: Security Extensions

Ana Maria de Alvare <anamaria@lll-lcc.llnl.gov> reports on the October 16-20, 1989 meeting in Brussels, Belgium:

The security working group worked the full week, discussing the draft. The meeting's primary goal was to approve the current draft for distribution to the international working groups. It was presented, at the EEC, to new members of the group from the European countries.

TRUSIX

A representative from TRUSIX, Charles Testa, gave a progress report on TRUSIX. [Editor's note – TRUSIX is an organization sponsored by the National computer security center (NCSC), developing a secure UNIX model. The participants are a number of vendors developing secure UNIX implementations.] Their modeling subcommittee has nearly completed a formal model describing the UNIX file system. They have accepted the "Ina Jo" model to describe Trusted UNIX System Interfaces. This model revolves around a MAC-read criterion, MAC-writes and a DAC constraint, and consists of simple security properties, confinement properties, and discretionary security properties representative of the Bell-LaPadula model. The TRUSIX ACL Rationale and Working Example Document has been approved by the NCSC and is being reviewed for publication under NCSC security publications.

One topic of interest to all security readers is prevention and/or detection of covert channels. TRUSIX is planning to include this under the Audit Rationale Document, which will include examples of typical covert channels and their implications. Issues such as bandwidth evaluation will be addressed by a separate white paper.

POSIX Conformance Testing

A representative from 1003.3, the POSIX Conformance Testing group, presented 1003.3's goal – to establish a series of specifications for testing the different POSIX standards. Although they have written the pseudo-code to test the conformance of a system to 1003.1, they feel they lack the staff and expertise to produce such tests for all the 1003 groups. Given this, their current plan is to draw upon each group for expertise and background knowledge of the subject at hand, and join those skills with their testing skills to produce a test bed for each 1003 standard.

Their ultimate goal is to allow testing of all elements of an open system for POSIX conformance by defining common test methods, which can then be implemented by private industries as test suites. They explained how to list the assertions, how to classify them, and what information they will need to produce a test method for 1003.6.

One factor mentioned was that the description has to address a single unit of behavior expected of a conformant system at a time. This implies dissecting interfaces into separate groups of assertions and generating assertions for both semantic and syntactic descriptions.

Discretionary Access Control (DAC)

The group focused on polishing and adding information to the draft. It was suggested the standard shouldn't define the behavior of *chmod* when old programs are being executed with the DAC mechanism.

It was noted that the current proposed Access Control List (ACL) might not be fully compatible with the current behavior of a *chmod* call. With the current, old-style behavior, when *chmod* is used to change owner, group and/or other permissions, the changed permissions represent the access modes of the file. In the current proposal for ACL, a *chmod* will provide the old behavior for the "owner" and "other" fields, but the "group" field permissions as set by *chmod* and shown by *stat*, wouldn't represent the actual access permissions of the group associated with that file; instead, it's a summary of all access permissions included under the ACL list for group entries. In other words, it would represent the maximum permissions allowed to the group entries included in the ACL list.

Some participants argued that *chmod* should be replaced by other system calls in a system conforming to 1003.6. In other words, if your system were to conform to 1003.6 the behavior of *chmod* would be unspecified and unpredictable.

I disagree. Although defining the behavior of *chmod* might restrict some implementations of ACLs, having a well-defined *chmod* behavior will provide backward compatibility and ease porting old programs to 1003.6-conformant systems. Otherwise old programs will have to be ported to platforms with system-dependent representations of *chmod* and *stat* information.

It was also proposed that the ACL list should allow entry types like timestamping. This would allow a policy that is more restrictive than the DOD orange-book policy to provide more granularity of file access.

Mandatory Access Control (MAC)

Kevin Murphy, of British Telecom, presented his views on electronic mail label usage and proposed that such a mechanism should be used as part of the standard. The electronic mail security labels consist of a generic format that includes four major components: security policy id, security classification, privacy mask, and security categories. This approach to labels is implemented on X.400 security services. One clear advantage of using such a format for labels is that it maximizes the potential synergy between operating system and electronic mail labels.

Chris Hughes from ICL presented British views on MAC information labels. Its main characteristics are these: object creation initializes the label, the label is implementation defined and changed by the owner, and the label is not used for access control. Chris proposed that the standard should provide a get/set mechanism for the object information label, and a way to merge and translate information labels, but should not standardize the labels' contents.

Information labels are useful because they provide added information on particular objects. We concluded that information labels should be in the scope of MAC as part of the standard, and requested that MITRE provide a presentation on information label use at the next meeting.

Privileges

The whole group heard a presentation of the internal draft of the privileges document. We decided that the wording had problems. The draft interface description is too obscure and needs a simpler description of privilege interfaces, before it can be included in the 1003.6 draft document.

Although the group argued considerably about the wording, they seemed to agree on the concepts. The main points are that privilege is associated with a process and privilege attributes can be attached to files.

I do not think I should burden the reader with the brainstorming ideas of the privilege group until a firmer position is taken at the next meeting. One thing I can say is that the process privilege concepts described in my last report (permitted, inheritable and effective) still stand, and a file still has three types of privilege attributes.

Audit

Kevin Brady from AT&T and Doug Steves from IBM presented a combined proposal, produced by them and Henry Hall from DEC, on how to define audit interfaces for 1003.6. Their proposal was meant to

contest the current audit stand, led by Olin Sibert from Oxford Systems.

The current audit definition is based on the token concept and on a pure procedural interface. In the procedural interface all data manipulation of the audit record is performed by function calls, with data passed explicitly through function parameters. Although this sounds attractive and clear, in practice it requires many function calls.

Another major point of controversy was the audit trail format. In Olin's proposal, conversion cost is minimal because writes and reads require an explicit specification of the format wanted. In Kevin, Doug, and Henry's proposal the conversion function is set to one of three conventional formats: little endian, big endian, or XDR. In other words, the information is stored in machine-dependent format while Olin's chooses a uniform format for all information stored.

One last contested feature was the ability to rewind audit trail information when viewing it. Kevin, Doug, and Henry's proposal does not allow a rewind, since information is manipulated at the data structure level.

Because of the heated discussion of procedural versus data structure interfaces for POSIX, both proposals will be formally written out, removed from the draft, and presented in the next meeting for a final vote.

Report on IEEE 1003.8/1: POSIX Transparent File Access

An anonymous correspondent reports on the July 10-14, 1989 meeting, in San Jose, California:

[Editor's note – Though this report came in substantially later than the other July reports, I think it's still useful, provocative, and well worth reading.]

Overview of New 1003.8 Developments

1. All standards produced by POSIX committees (with the exception of language-binding standards like 1003.5 and 1003.9) must be specified in a language-independent fashion, and must include at least one language-specific binding. Since P1003 will not have guidelines and rules for constructing a language-independent specification before April 1990, no POSIX networking group can possibly ballot a document before July 1990. "Mock" ballots (aka trial-use ballots) are unaffected by this, but their usefulness will be diminished.

2. Two new POSIX Networking working groups either have submitted or will soon submit PARs to begin work, bringing the total number of POSIX Networking working groups to six. These new groups are the Name Space and Directory Services Interface and the X.400 Mail Gateway Interface. [Editor's note – The SEC approved the PAR for the former, but decided that the latter transcends POSIX, and recommended that the IEEE form a separate, numbered group, analogous to 1003 or 1201, to handle X.400. See below.]

3. Due to the rules governing IEEE and IEEE-TCOS standards activities, as well as the logistical nightmare six sub-working groups cause, the hierarchical structure of the P1003.8 POSIX networking committee will be flattened out, with each current sub-group submitting PARs to cover their current work. Coordination will be provided by a "POSIX Networking Steering Committee," made up of the chairs and vice-chairs of each networking-related working group and the current chair and vice-chair of 1003.8. [Editor's note – This is still being debated by the SEC.]

4. Since each of the 1003.8 sub-groups will be submitting separate PARs, the P1003 Sponsor's Executive Committee (SEC) is taking the opportunity to evaluate the degree to which each PAR is intended to represent a part of the "POSIX Environment" or is a component which has no relationship to the rest of POSIX and should, hence, stand alone. The result of this is that several of the 1003.8 sub-groups may be issued project numbers outside of the P1003 family. There is some precedent for this; the X11 interface group was assigned project number P1201 for just this reason.

Activity in the TFA Subgroup, P1003.8/1

The group is making slow but steady progress towards the goals of a fully-specified programmatic interface for networked file

access and the semantics and suggested syntax for administrative interfaces for such a functionality. The group is dominated by a faction, apparently led by Sun Microsystems, with a goal of ensuring that NFS, in some form, is a sufficient mechanism to provide the service required by the "full TFA" interface. The balance of the committee is composed of people who simply want a standard they can use as an acquisition tool.

Achievements

• Enough consensus and material was obtained to permit development of a first draft of the programmatic interface part of the specification; this draft should be complete in time for the second mailing, due out on September 8.

• Liaison activities with 1003.7 (System Administration) continued; .7 indicated that all of the options for the TFA *mount / unmount* model were, in fact, of use in administering such a system. They also agreed that they owned responsibility for all file-system commands not completely unique in function to TFA, and that they would pursue input from the TFA group when the time was right.

• Further progress was made on identifying status and usage information which must be obtainable from the provider of a TFA service.

Problem Areas

1. Representation in the group is unbalanced; there is, as of this time [Editor's note – July, '89], no substantial representation of the "stateful" side of the semantic issues. The chairman has, so far, been unsuccessful in encouraging a more balanced group viewpoint so representation from the stateful camp has been solicited (with minimal success) for future meetings.

2. Several "grey areas" in the semantics of IEEE 1003.1-1988 have been identified by the TFA group over the last several months. The suggested "fixes" have been slanted in a way that would let the TFA interface avoid relaxing 1003.1 semantics while permitting a stateless implementation of the TFA service; i.e. rather than strengthening 1003.1 to define the actual behavior of a single stand-alone system, the proposals have been so weak that they under-constrain single-system behavior. It appears that the majority of the 1003.1 committee will not approve of this approach, and will require the "fix" to be of the proper strength to correctly specify single-system behavior.

Let me give an example. The original 1003.1 standard is silent on the issue of when the effects of a write are visible to a subsequent read of the same byte of the file. If process A writes byte 123 of file *foo*, then process B does a read of byte 123 of file *foo*, at what point is B guaranteed to see the byte A wrote?

Immediately? If so, stateless solutions employing read caches fail; if process B is remote on system "bsys" and reading the file via NFS, byte 123 might come out of the file cache on bsys and not from the file cache on the system where A lives.

Immediately if A and B are on the same system, and at some implementation-defined time otherwise? This requires 1003.1 to define what it means by "the same system," and doesn't adequately address multi-processor single systems with "interesting" caches. It also means a truly portable application that is interested in running in a distributed environment can **never** know when the byte written by A is visible to B.

Only in the presence of byte locking? In other words, A locks byte 123, writes it, unlocks it; if B then locks and reads 123, it is guaranteed to see what A wrote. Not a bad solution, but it breaks existing applications and in fact is a relaxation of the intended semantics of 1003.1.

Basically, the "intent" developing in 1003.1 is that the effect of the write must be seen immediately by *any other process with that file open,* without regard to process location, without recourse to O_SYNC mode opens, without the necessity for locking, and so on. 1003.1-1988 is silent on the issue; the proposed fix from TFA (basically a compromise I did not like and knew would fail) was that read-after-write be guaranteed only for co-located processes and in the presence of locking. This gave 1003.1 a chance to avoid relaxing their intended semantics while leaving TFA a "loophole" to change semantics

without having to indicate a change in wording from 1003.1.

This is what got rejected by 1003.1, which is getting pretty damned tired of TFA's trying to claim that the full TFA semantics are "just like" 1003.1 but with gaping differences that are introduced silently due to weak or weasel wording in 1003.1-1988.

3. 1003.7, System Administration, is making distressingly slow progress. If this continues, 1003.8 will have two choices with respect to client-side administrative commands:

- Do not standardize them; give feedback to 1003.7 and wait for them to complete their specification. This risks incompatible implementations.

- Standardize them insofar as they relate to TFA administration. This risks incompatibility between the TFA aspects of those commands and their more general aspects. An example is the *mount* command; if 1003.7 is unhappy with the form of the TFA mount command, they are under no constraint to remain compatible with it. If the group ballots far enough in advance of 1003.7, this sort of clash could be frequent.

4. Many of the contentious issues have been "resolved" through the various mechanisms POSIX provides for introducing optional behavior; most frequently, these involve either "implementation-defined" behavior, or the addition of path-specific attributes whose status can be determined through the *pathconf()* function. Several of these options should be viewed by the ballot group as being "gratuitous" in some sense; i.e., the TFA committee should take a stand one way or another, and be willing to be beaten up in ballot for it. The POSIX standards are wishy-washy enough without the addition of gratuitous options.

Report on IEEE 1201: User Interface

Eileen Coons <coons@osf.org> reports on the October 16-19, 1989 meeting in Brussels, Belgium:

"The time has come," the walrus said,
"To talk of many things:
Of shoes – and ships – and sealing wax –
Of cabbages – and kings –
And why the sea is boiling hot –
And whether pigs have wings."
– Lewis Carroll

The P1201 committee is on a divine mission to define standards for user interface technologies. Lewis Carroll would have loved P1201 meetings.

In keeping with the precedent set by previous P1201 meetings, this latest get-together was spirited. The quasi-good news is that, by the end of the session, not one, but three PARs had been defined, as the group split into 1201.1 (Application Programming Interface), 1201.2 (Drivability - Look & Feel), and 1201.3 (User Interface Definition Language). One participant aptly named the proceedings "PAR Wars."

There was agonized discussion over the various sub-groups' missions, and an equal amount of agonized, and at times agonizing, wordsmithing over the .1 and .2 PARs themselves. The .3 group thoughtfully elected to split off and define itself in private. The PARs will be submitted via proper official channels to be blessed at the January SEC meeting.

For anyone not familiar with the PAR process, PAR is an acronym for Project Authorization Request. An individual or group that believes some work should be done by an IEEE committee drafts a document describing the work, which is then submitted to the IEEE as a PAR. Usually the PAR is circulated to the IEEE membership.

The Standards Executive Committee (SEC) reviews the PAR during its next scheduled session, typically held during a POSIX meeting. The SEC votes on the PAR, and if it is approved by the SEC, it is presented to Technical Committee on Operating Systems (TCOS). TCOS decides in which committee the work will be done. In the case of the PAR for User Interface, TCOS elected to divorce the work from the core POSIX effort (1003), and created P1201.

The PAR becomes part of the statement of work and basic charter for the group doing the work.

Fortunately, at this meeting, the group finally created some real structure for itself. The group decided to define an agenda and resolved that all meeting attendees should receive minutes of the meeting. Jim Isaak, the chair of the 1003 SEC, helped with structural definition by supplying IEEE rules and charter information, explaining the balloting process, and listing action options open to the committee.

Seven ballot alternatives were proposed, ranging from submitting a proposal for immediate ballot, to disbanding 1201. A vote was called, and although there was no consensus the heavy favorite was a proposal to adopt Motif's API as the basis for a standard API specification, and to extend it to accommodate aspects of Open Look's style.

This general direction was unpopular with a vocal minority, and so the group discarded the vote and returned to its original, pre-poll path of action: defining a specification for an API based on neither Motif nor Open Look, but on some new API – probably a hybrid of the two.

[Editor's note: I've heard more than one person express ill-ease about the restricted range of choices being considered. Why is there no mention of NeXT/Step, for example? A noticeable feeling among people who aren't on the committee is that it's too early to try to standardize in this area, and that the answer to the question, "Motif or Open Look?" should be, "No thanks." The answer to the implied question, "Why is there a P1201 and why are we doing this now, anyway?" seems to be that NIST, the National Institute for Standards and Technology (the people who bring you FIPS), is pushing hard for rapid creation of a GUI standard.]

Two presentations were made: one by AT&T, in favor of the joint API concept, and one by OSF, arguing against its feasibility. This was followed by a critique of – some thought, attack on – the second presentation by one of the acting chairs, Clive Feather of X/OPEN. P1201 may be many things but, so far, staid isn't one of them...

On a more neutral note, several representatives from organizations working on UIDL technologies made presentations about what they were doing in that arena, and then went off to form P1201.3.

The rest of the group broke into the .1 and .2 sub-groups for working sessions during most of the remaining meeting time. Each group reviewed its newly drafted PAR. P1201.1 also spent time comparing Motif and Open Look, identifying and exploring the differences between the two API's, and looking for potential drivability issues that could be deferred to P1003.2. P1003.2 took a similar course of action, comparing the styles of the two technologies.

There was also a spirited discussion regarding when and where the next P1201 meetings should be held. After various alternatives were explored, the group decided to keep P1201 meetings in the same vicinity and timeframe as POSIX meetings, since many attendees need, or want, to participate in POSIX as well.

USENIX ASSSOCIATION

ORDER FORM

CONFERENCE & WORKSHOP PROCEEDINGS

Qty	Proceeding			Add'l Unit Price	Total	Foreign Postage	Total
___	Washington DC Conference	Jan.	'90	$25	$_____	$15	$_____
___	Graphics Workshop V	Nov.	'89	18	$_____	10	$_____
___	Distributed and Multiprocessor Workshop	Oct.	'89	30	$_____	20	$_____
___	Large Installation Sys. Admin. III Workshop	Sept.	'89	13	$_____	9	$_____
___	Baltimore Conference	June	'89	20	$_____	15	$_____
___	UNIX Transaction Processing Workshop	May	'89	12	$_____	8	$_____
___	Software Management Workshop	Apr.	'89	20	$_____	15	$_____
___	San Diego Conference	Feb.	'89	30	$_____	20	$_____
___	C++ Conference	Oct.	'88	30	$_____	20	$_____
___	C++ Workshop	Nov.	'87	30	$_____	20	$_____
___	Graphics Workshop IV	Oct.	'87	10	$_____	15	$_____
___	Washington DC Conference	Jan.	'87	10	$_____	20	$_____
___	Graphics Workshop III	Dec.	'86	10	$_____	15	$_____

Total price of Proceedings _____
Calif. residents only add applicable sales tax _____
Total foreign postage _____
Total enclosed _____

* Discounts are available for bulk orders. Please inquire.

PAYMENT OPTIONS

☐ Check enclosed payable to USENIX Association. ☐ Purchase order enclosed.
☐ Please charge my: ☐ Visa ☐ MasterCard

Account # _____ Exp.Date _____

Signature _____

Overseas? Please make your payment in U.S. currency by one of the following:
* Charge (Visa, MasterCard, or foreign equivalent)
* International postal money order
* Check - issued by a local branch of a U.S. Bank

1/90

Shipping Information

Orders to U.S. and Canada are shipped via printed matter. Please allow 2 weeks for delivery. Foreign orders are shipped via air printed matter. Please allow 10-14 days for delivery. Shipment by UPS, first class, or airmail is available upon request.

Ship to: _____

Please mail this order form to: USENIX Association
2560 Ninth Street
Suite 215
Berkeley, CA 94710

4.3BSD MANUAL REPRODUCTION AUTHORIZATION and

ORDER FORM

Purchase Order #: _____

Date: _____

Pursuant to the copyright notice as found on the rear of the cover page of the UNIX® /32V Programmer's Manual stating that:

> "Holders of a UNIX®/32V software license are permitted to copy this document, or any portion of it, as necessary for licensed use of the software, provided this copyright notice and statement of permisssion are included,"

I hereby appoint the USENIX Association as my agent, to act on my behalf to duplicate and provide me with such copies of the Berkeley 4.3BSD Manuals as I may request.

Signature: _____

The prices below include shipping (via UPS) and handling charges for domestic orders. Overseas postage rates will be provided upon request.

4.3BSD User's Manual Set (3 vols)	_____ at $28.00 each	= $_____
4.3BSD Programmer's Manual Set (3 vols)	_____ at $28.00 each	= $_____
4.3BSD System Manager's Manual (1 vol.)	_____ at $12.00 each	= $_____
	Total	= $_____

Discounts are available for bulk orders. Please inquire.

PAYMENT OPTIONS

☐ Check enclosed payable to USENIX Association. ☐ Purchase order enclosed. Please send an invoice.

☐ Please charge my: ☐ Visa ☐ MasterCard

Account # _____ Expiration Date _____

Signature _____

1/90

Ship to: _____ Bill to, if different: _____

Phone #: _____

Mail your check or purchase order with this order form to:

USENIX Association, Manual Order Dept.
2560 Ninth Street, Ste. 215
Berkeley, CA 94710
415/528-8649

PLEASE ALLOW 2 WEEKS FOR RECEIPT OF YOUR ORDER.

Local User Groups

The Association will support local user groups by doing a mailing to assist the formation of a new group and publishing information on local groups in *;login:*. At least one member of the group must be a current member of the Association. Send additions and corrections to *login@usenix.org*.

CA – Fresno: the Central California UNIX Users Group consists of a *uucp*-based electronic mailing list to which members may post questions or information. For connection information:

Educational and governmental institutions:
Brent Auernheimer (209) 294-4373
brent@CSUFresno.edu or csufres!brent

Commercial institutions or individuals:
Gordon Crumal (209) 875-8755
csufres!gordon (209) 298-8393

CO – Boulder: the Front Range UNIX Users Group meets monthly at different sites.

Steve Gaede (303) 447-8586
636 Arapahoe Ave., #10
Boulder, CO 80302

FL – Coral Springs:

S. Shaw McQuinn (305) 344-8686
8557 W. Sample Road
Coral Springs, FL 33065

FL – Fort Lauderdale/Miami: The South Florida UNIX Users Group meets the 2nd Tuesday of each month.

Tony Vincent, John McLaughlin (305) 776-7770
{sun,novavax,gould}!sunvice!tony
jmclaughlin@sun.COM

John O'Brien (305) 475-7633
gatech!uflorida!novavax!john

Don Joslyn (305) 476-6415
gatech!uflorida!novavax!rm1!don

FL – Jacksonville/Northeast: UNIX Users of Jacksonville (uujax) meets the 2nd Thursday of each month.

Tom Blakely (904) 646-2820
uflorida!unf7!tfb

Emilie Olsen (904) 390-3621

FL – Melbourne: the Space Coast UNIX Users Group meets at 8pm on the 3rd Wednesday of each month at the Florida Institute of Technology.

Bill Davis (407) 242-4449
bill@ccd.harris.com

FL – Orlando: the Central Florida UNIX Users Group meets the 3rd Thursday of each month.

Mike Geldner (305) 862-0949
codas!sunfla!mike

Ben Goldfarb (305) 275-2790
goldfarb@hcx9.ucf.edu

Mikel Manitius (305) 869-2462
{codas,attmail}!mikel

FL – Tampa Bay: the Tampa UNIX Users Group meets the 1st Thursday of each month in Largo.

Bill Hargen (813) 530-8655
uunet!pdn!hargen

George W. Leach (813) 530-2376
uunet!pdn!reggie

GA – Atlanta: meets on the 1st Monday of each month in White Hall, Emory University.

Atlanta UNIX Users Group
P.O. Box 12241
Atlanta, GA 30355-2241

Marc Merlin (404) 442-4772
Mark Landry (404) 365-8108

MI – Detroit/Ann Arbor: The SouthEastern Michigan Sun Local Users Group meets jointly with the Nameless UNIX Group on the 2nd Thursday of each month in Ann Arbor.

Steve Simmons home: (313) 426-8981
scs@lokkur.dexter.mi.us office: (313) 769-4086

K. Richard McGill
rich@sendai.ann-arbor.mi.us

Bill Bulley
web@applga.uucp

MI – Detroit/Ann Arbor: dinner meetings the 1st Wednesday of each month.

Linda Mason (313) 855-4220
michigan!/usr/group
P.O. Box 189602
Farmington Hills, MI 48018-9602

MN – Minneapolis/St. Paul: meets the 1st Wednesday of each month.

UNIX Users of Minnesota
17130 Jordan Court
Lakeville, MN 55044

Robert A. Monio (612) 895-7007
pnessutt@nis.mn.org

MO – St. Louis:

St. Louis UNIX Users Group
Plus Five Computer Services
765 Westwood, 10A
Clayton, MO 63105

Eric Kiebler (314) 725-9492
plus5!sluug

NE – Omaha: meets the 2nd Thursday of each month.

/usr/group nebraska
P.O. Box 44112
Omaha, NE 68144

Kent Landfield (402) 291-8300
kent@ugn.uucp

New England – Northern: meets monthly at different sites.

Peter Schmitt (603) 646-2999
Kiewit Computation Center
Dartmouth College
Hanover, NH 03755

decvax!dartvax!nneuug-contact

NJ – Princeton: the Princeton UNIX Users Group meets monthly.

Pat Parseghian (609) 452-6261
Dept. of Computer Science
Princeton University
Princeton, NJ 08544

pep@Princeton.EDU

NY – New York City:

Unigroup of New York
G.P.O. Box 1931
New York, NY 10116

Ed Taylor (212) 513-7777
{attunix,philabs}!pencom!taylor

OK – Tulsa: the Tulsa UNIX Users Group, $USR, meets the 2nd Wednesday of each month.

Stan Mason (918) 560-5329
tulsix!smason@drd.com

Mark Lawrence (918) 743-3013
mark@drd.com

PA – Philadelphia: the UNIX SIG of the Philadelphia Area Computer Society (PACS) meets the morning of the 3rd Saturday of each month.

G. Baun, UNIX SIG
c/o PACS
Box 312
La Salle University
Philadelphia, PA 19141

rutgers!{bpa,cbmvax}!
temvax!pacsbb!{gbaun,whutchi}

TX – Dallas/Fort Worth:

Dallas/Fort Worth UNIX Users Group
Seny Systems, Inc.
5327 N. Central, #320
Dallas, TX 75205

Jim Hummel (214) 522-2324

TX – Houston: the Houston UNIX Users Group (Hounix) meets the 3rd Tuesday of each month.

Hounix answering machine (713) 684-6590

Bob Marcum, president (713) 270-8124

Chuck Bentley, vice-president (713) 789-8928
chuckb@hounix.uucp

TX – San Antonio: the San Antonio UNIX Users (SATUU) meets the 3rd Thursday of each month.

Jeff Mason (512) 494-9336
Hewlett Packard
14100 San Pedro
San Antonio, TX 78232

gatech!petro!hpsatb!jeff

WA – Seattle: meets monthly.

Bill Campbell (206) 232-4164
Seattle UNIX Group Membership Information
6641 East Mercer Way
Mercer Island, WA 98040

uw-beaver!tikal!camco!bill

Washington, D.C.: meets the 1st Tuesday of each month.

Washington Area UNIX Users Group
2070 Chain Bridge Road, Suite 333
Vienna, VA 22180

Samuel Samalin (703) 448-1908

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710

First Class Mail

FIRST CLASS MAIL
U.S. POSTAGE PAID
Hayward, CA
Permit No. 2

1989 Membership Survey Results

Professional Development Seminars

Speakers Bureau

Book Reviews and much, much more ....

Change of Address Form

Please fill out and send the following form through the U.S. mail to the Association Office at the address above.

Name: _____ _____ Member #: _____

OLD: ___ _____ NEW: _____ _____

_____ _____

_____ _____

_____ _____

Phone: _____ ___

uucp: _____ _____